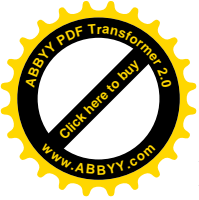


**МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ**

**"ОСНОВИ WEB-ДИЗАЙНУ"
МЕТОДИЧНІ ВКАЗІВКИ І ЗАВДАННЯ
ДО САМОСТІЙНИХ І ЛАБОРАТОРНИХ РОБІТ
(для студентів економічних спеціальностей)**

Донецьк - 2012



**МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ І СПОРТУ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ**

**"ОСНОВИ WEB-ДИЗАЙНУ"
МЕТОДИЧНІ ВКАЗІВКИ І ЗАВДАННЯ
ДО САМОСТІЙНИХ І ЛАБОРАТОРНИХ РОБІТ**

(для студентів економічних спеціальностей)

*Розглянуто на засіданні кафедри
обчислювальної математики і
програмування.
Протокол № 9 від 09.04.2012 р.*

*Затверджено на на навчально -видавничій
раді ДонНТУ.
Протокол № 2 від 19.04.2012 р.*

Донецьк - ДонНТУ - 2012



УДК 002(075.8)
ББК 32.81я73

"Основи WEB-дизайну". Методичні вказівки і завдання до самостійних і лабораторних робіт (для студентів економічних спеціальностей) /укладачі: Д.В. Бельков, Є.М. Єдемська – Донецьк: ДонНТУ, 2012. – 127 с.

Навчально-методичний посібник відповідає програмі навчального курсу дисципліни „Інформатика”. Він містить необхідний теоретичний матеріал та завдання для виконання студентами економічних спеціальностей лабораторних робіт за темою „Основи WEB-дизайну”.

Методичні вказівки призначені для студентів денної і заочної форми навчання.

Укладачі:

Д.В. Бельков, доцент,
Є.М. Єдемська, ст. викладач,

Рецензент:

Анохіна І.Ю., доцент

Відповідальний за випуск:

В.М. Павлиш, професор



ЗМІСТ

ВСТУП.....	4
1 ОСНОВИ ВЕБ-ДИЗАЙНУ.....	5
1.1 Коротка характеристика мови HTML.....	5
1.2 Визначення загальних параметрів.....	7
1.3 Форматування тексту.....	10
1.4 Використання тегів заголовків.....	13
1.5 Форматування HTML-документу.....	15
1.6 Використання коментарів та спеціальних символів.....	17
1.7 Рисунки.....	18
1.8 Звук.....	22
1.9 Гіперпосилання.....	23
1.10 Списки.....	24
1.10.2 Нумеровані списки.....	26
1.10.3 Списки визначень.....	26
1.11 Таблиці.....	28
1.12 Фрейми.....	29
2 ЛАБОРАТОРНИЙ ПРАКТИКУМ.....	32
2.1 Лабораторна робота № 1. "Використання ресурсів мережі Інтернет за допомогою браузеру Microsoft Internet Explorer".....	32
2.2. Лабораторна робота №2 "Створення текстової HTML-сторінки".....	36
2.3. Лабораторна робота №3 "Використання рисунків, відео та звуку на HTML-сторінці".....	57
2.4. Лабораторна робота №4 "Створення гіперпосилань".....	66
2.5. Лабораторна робота №5 "Створення списків".....	75
2.6. Лабораторна робота №6 "Створення таблиць ".....	93
2.7 Лабораторна робота №7 "Створення фреймів ".....	114
ЛІТЕРАТУРА.....	126



ВСТУП

Ефективне управління виробництвом, формування оптимальних рішень у різноманітних сферах економіки та бізнесу, сталий розвиток підприємства в значній мірі залежать від рівня впровадження та використання інформаційних систем (ІС) і технологій. На сучасному етапі розробка та впровадження досконалих ІС неможлива без їх адаптації до використання в глобальній мережі Internet, а також однієї з її основних частин мережі WWW. Характерною рисою адаптації є наявність в ІС Web-сайту, доступного користувачам завдяки мережі WWW. Широке розповсюдження Web-сайтів призвело до появи нового напрямку в галузі програмування – Web-дизайну.

В сучасних умовах невеликі компанії нерідко створюють та підтримують Web-сайти силами своїх працівників. Крім того, досвід впровадження сучасних складних Web-орієнтованих систем свідчить про необхідність розробки подібних систем шляхом активної взаємодії між компанією-розробником програмного забезпечення та представниками замовника таких систем. Для цього необхідно знати принципи проектування та розробки Web – сайтів, мати чітке уявлення про їх структуру, принципи функціонування, місце і роль в організації роботи підприємства, мати уявлення про перспективи подальшого розвитку Web-систем, а також вміти їх використовувати на високому рівні.

Таким чином, в сучасних умовах фахівці різного профілю повинні опанувати Web-дизайн. Відзначимо, що на сьогодні по причині своєї новизни напрям Web-дизайну не має чітких меж. Досить часто до цього напрямку відносять розробку серверного та клієнтського програмного забезпечення, адміністрування Web-сайту, реалізацію заходів по підвищенню ефективності його функціонування та формування власне самого дизайну Web-сайту. Проте, як свідчить практика, основним завданням Web-дизайнера є розробка та підтримка клієнтського програмного забезпечення Web-сайту для забезпечення високої ефективності його функціонування. Всі інші завдання потребують достатньо глибоких професійних знань та навиків, і вирішуються спеціалістами відповідного профілю.

Гіперпосилання, основним призначенням яких є зв'язування між собою Web-ресурсів, взагалі є базисом побудови мережі WWW. Таблиці використовуються в якості основного засобу форматування HTML-документів. Це пов'язане з тим, що розміщення інформації в HTML-документі виконане за допомогою інших засобів має або набагато менші можливості, або не однаково відображається браузерами різних типів. В свою чергу списки є широко розповсюдженою формою показу структурованих даних. Застосування списків може значно зменшити трудомісткість форматування таких даних.

Сфера застосування фреймів має певну специфіку. Відзначимо, що в сучасних стандартах побудови HTML-документів використовувати фрейми не рекомендується, в основному по причині труднощів обробки таких документів пошуковими системами. Проте фрейми досить широко вживаються для HTML-документів, що публікуються в локальних мережах, або розповсюджуються на магнітних та оптичних дисках.



1 ОСНОВИ ВЕБ-ДИЗАЙНУ

1.1 Коротка характеристика мови HTML

HTML-документ – файл текстової або нетекстової природи (звук, відео, зображення), створений за допомогою *мови гіпертекстової розмітки HTML* (Hyper Text Mark-up Language).

Застосування універсальної мови HTML зумовлене необхідністю представлення інформації для глобального використання комп'ютерами різноманітних типів, що працюють під управлінням різноманітних операційних систем. Існують стандарти мови HTML, які підтримуються практично всіма провідними компаніями-розробниками програмного та апаратного забезпечення. Ці стандарти передбачають, що HTML-документ буде однаково та коректно представлений будь-якому користувачеві мережі WWW.

Мова HTML надає розробникам засоби для:

- Публікації HTML-документів з заголовками, текстом, таблицями, списками, рисунками і т.і.
- Навігації по HTML-документам за допомогою гіперпосилань.
- Розробки форм для обміну даними з відділеними службами, що можуть бути використані при пошуку інформації, в електронній комерції і т.і.
- Включення електронних таблиць, відео та аудіо кліпів та інших додатків безпосередньо в HTML-документи.

Текстовий HTML-документ представляє собою звичайний текстовий файл з розширенням HTML або HTM. Відзначимо, що в назві файлу можливо використовувати тільки символи латинського алфавіту, арабські цифри, символи дефісу ("-") та підкреслення ("_").

В цьому документі може міститись інформація двох типів: звичайний текст та команди (*теги*) мови HTML. Зміст HTML-документу інтерпретуються браузером під час завантаження. Як правило, наслідком інтерпретації тексту є його відображення у вікні браузера. Результати інтерпретації тегів залежать від їх призначення. Наприклад, це може бути форматування тексту або відображення у вікні браузера певного рисунку.

Всі теги мови HTML-документу виділяються обмежуючими символами (< та >), між якими записується *ідентифікатор (ім'я) тегу*. Розрізняють *одинарні* та *парні теги*.

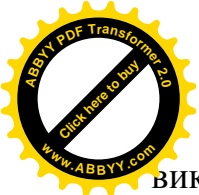
Наведемо приклад одинарного тегу для відображення у вікні браузера горизонтальної лінії <hr>.

На відміну від одинарного для визначення парного тегу (*тегу-контейнеру*) використовуються *відкриваючий* та *закриваючий тег*. Закриваючий тег відрізняється наявністю символу / перед іменем тегу.

Наприклад:

```
<td> Зміст тегу </td>
```

Все, що записано між відкриваючим та закриваючим тегом називають *змістом тегу*. В деяких випадках закриваючий тег можливо не



використовувати. Наприклад тег–контейнер `` можливо використовувати без закриваючого тегу ``.

Теги можуть мати **параметри (атрибути)**. Наприклад, для визначення розміру тексту в тезі `` можливо використати атрибут `size`:

```
<font size='12'>
```

Кількість та номенклатура параметрів індивідуальні для кожного тегу. Якщо параметрів декілька, то вони відокремлюються між собою за допомогою пробілів. Наприклад:

```
<font size=12 color='red'>
```

Відзначимо, що як імена тегів, так і імена параметрів не чутливі до регістру символів, за допомогою яких вони визначені. Таким чином, визначення тегів `
` та `
` рівнозначне. Розрізняють параметри тегів, які потребують запису значення та параметри, які використовуються без значення.

Якщо параметр тегу не потребує запису значення, то використовується тільки назва параметру. Наприклад:

```
<br noshade>
```

Якщо параметр тегу потребує запису значення, то воно вказується після назви параметру після символу `"=`". Значення параметру може записуватись як в одинарних або подвійних лапках, так і взагалі без лапок. Наприклад:

```
<td align='left'>
```

```
<td align="left">
```

```
<td align=left>
```

Проте рекомендується записувати значення параметру в одинарних або подвійних лапках.

Крім одинарних та парних тегів розрізняють **теги рівня блоку** та **послідовні теги**. Різниця між ними полягає в тому, що теги рівня блоку можуть містити послідовні теги та інші теги рівня блоку. Послідовні теги можуть містити тільки данні та інші послідовні теги. Наведемо приклади:

- Теги рівня блоку: `<table></table>`, `<p></p>`.
- Послідовні теги: ``, `
`.

Важливою особливістю мови HTML є те, що неправильно записані теги або параметри тегів ігноруються браузером. При цьому браузер не показує ніякого повідомлення та переходить до інтерпретації інших частин HTML-документу.

Мова HTML передбачає можливість використання кольорового форматування HTML-документу. При цьому кольори визначаються або за допомогою англійських назв, або в форматі **RGB** (*Red Green Blue*).

Приклади визначення кольорів за допомогою англійських назв:

- *green* – зелений.
- *red* – червоний.
- *blue* – синій.
- *white* – білий
- *black* – чорний.



Хоча такий спосіб визначення кольору досить зручний, але він обмежений кількістю відповідних назв. Тому більш поширений є спосіб визначення кольорів у форматі RGB, який дозволяє використовувати більш ніж 1,5 млн. кольорів.

Передумовою використання формату RGB є те, що будь-який колір можливо визначити за допомогою трьох кольорів: червоного (red), зеленого (green) та синього (blue). В форматі RGB кожному із цих кольорів може відповідати число від 0 до 256, записане в шістнадцятичному форматі. Відсутність даного кольору відповідає числу 0 (00), а максимальна насиченість числу 256 (ff). Синтаксис RGB передбачає використання знаку "#".

Приклади визначення кольорів в форматі RGB:

- Червоний – #ff0000.
- Зелений – #00ff00.
- Синій – #0000ff.
- Чорний – #000000.
- Білий – #ffffff.

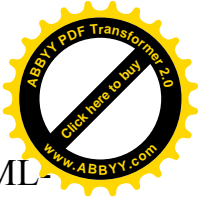
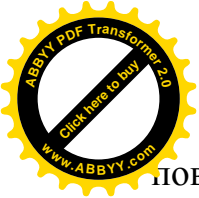
Для створення HTML-документу достатньо використовувати текстовий редактор, який не вставляє в текст документу власних команд. Прикладом може бути редактор "Блокнот", що входить в стандартну комплектацію операційної системи Windows.

Технологія створення HTML-документу в випадку застосування "Блокноту" полягає в тому, що Web-програміст визначає текст HTML-документу та теги, що в ньому використовуються. Після цього HTML-документ зберігається у файлі формату html або htm та тестується за допомогою браузеру. Після виправлення помилок HTML-документ може бути розміщений на Web-сервері.

Хоча вказана технологія є досить простою та не потребує спеціального програмного забезпечення, але вона є не оптимальною з точки зору трудомісткості, виявлення помилок при визначенні тегів та високих вимог до кваліфікації Web-програміста. Високі вимоги пояснюються тим, що Web-програмісту необхідно пам'ятати великий обсяг інформації, пов'язаний з номенклатурою та синтаксисом тегів. Тому на сучасному етапі використовують спеціальні HTML-редактори, які дозволяють суттєво підвищити якість програмного коду та прискорити процес створення HTML-документів. Найбільш популярними серед вітчизняних Web-програмістів є HTML-редактори – Macromedia Homesite та Macromedia Dreamweaver. Популярність цих редакторів пояснюється дружнім інтерфейсом користувача, наявністю вбудованої в редактори бібліотеки тегів та синтаксичного аналізатора правильності програмного коду HTML-документу, а також можливістю застосування візуальної технології створення HTML-документів.

1.2 Визначення загальних параметрів

Першим тегом, з якого слід починати опис HTML-документу, є тег <html>. Він завжди повинен починати опис документу, а завершати опис



повинен тег `</html>`. Тобто тег `<html>` є тегом контейнером для всієї HTML-сторінки. Сам документ є звичайним текстовим ASCII-файлом і може складатися із двох розділів – заголовка та змістовної частини.

Розділ заголовка відкривається тегом `<head>` і закривається тегом `</head>`. Задачею заголовку є надання браузеру інформації, необхідної для коректного відображення HTML-сторінки. Теги, що знаходяться всередині розділу заголовку (крім назви документу), на екрані не відображаються. Назву документу визначають за допомогою тегу контейнеру `<title>`. Наприклад:

```
<title> Назва HTML-сторінки </title>
```

Змістова частина документу, що не містить фреймів, визначається тегом контейнером `<body>`. Тобто весь текст та всі теги змістовної частини документу знаходиться між тегами `<body>` та `</body>`.

Таким чином, для визначення простої HTML-сторінки досить написати:

```
<html><head><title> Заголовок моєї Web-сторінки</title>  
</head><body>  
Зміст Web-сторінки  
</body></html>
```

Відображення цієї сторінки в вікні браузера показано на рис. 1.1.

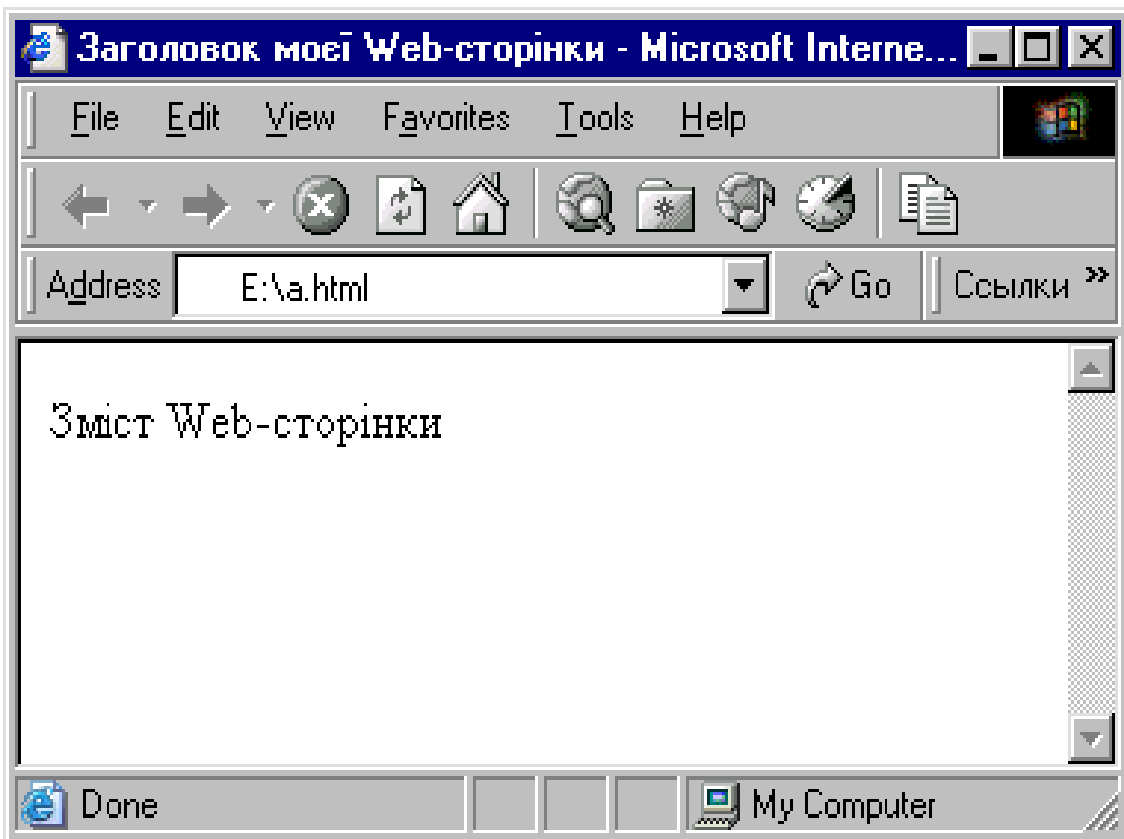
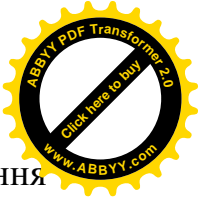


Рис.1.1 - Відображення простої HTML-сторінки браузером

Відзначимо, що в розділі заголовку досить часто використовується тег `<meta>`. За допомогою цього тегу вирішують деякі типові задачі:



1. Визначення кодування тексту HTML-сторінки. Наприклад, визначення того, що кодування тексту HTML-сторінки, задане в найпоширенішому серед вітчизняних користувачів форматі windows-1251, можливо здійснити так:

```
<META http-equiv=Content-Type content="text/html; charset=windows-1251">
```

2. Визначення ключових слів, що використовуються пошуковими системами. Наприклад, визначення ключових слів "економіка" та "торгівля" можливо здійснити так:

```
<META name="keywords" content="економіка, торгівля">
```

3. Визначення опису HTML-сторінки в пошуковій системі. Наприклад:

```
<META name="description" content = "сайт, присвячений економіці та торгівлі ">
```

4. Відображення в браузері визначеної HTML-сторінки. Наприклад, для переходу на HTML-сторінку 1.html через 20 секунд після відображення поточної HTML-сторінки необхідно:

```
<meta http-equiv="refresh" content="20;URL=untitled1.html">
```

Також необхідно відзначити, що тег <body> може мати ряд необов'язкових параметрів (див. табл.1.1), які визначають форматування всієї HTML-сторінки.

Таблиця 1.1- Параметри тегу <body>

Параметр	Призначення
alink	Визначає колір активного гіперпосилання
background	Вказує на адресу фонового зображення
bottommargin	Визначає межу нижнього поля документу
bgcolor	Визначає колір фону документу
bgproperties	Дозволяє або забороняє прокрутку фонового зображення
leftmargin	Визначає межу лівого поля документу
link	Визначає колір не переглянутого гіперпосилання
rightmargin	Визначає межу правого поля документу
scroll	Встановлює або забороняє смуги прокрутки вікна браузера
text	Визначає колір тексту
topmargin	Визначає межу верхнього поля документу
vlink	Визначає колір переглянутого гіперпосилання

Наприклад, для визначення документу з текстом зеленого кольору та відсутністю смуг прокрутки слід написати:

```
<body text="green" scroll="0">
```



1.3 Форматування тексту

Форматування тексту засобами HTML реалізується за допомогою спеціальних тегів. Всі теги форматування тексту використовуються тільки в розділі <body> і є парними тегами. Найчастіше використовуються такі теги:

Тег – призначений для відображення тексту напівжирним шрифтом.

Наприклад:

Це напівжирний шрифт

Тег <i> – призначений для відображення тексту курсивом.

Наприклад:

<i>Це курсив</i>

Тег <u> – призначений для відображення тексту підкресленим.

Наприклад:

<u>Це підкреслений текст</u>

Тег <s> – призначений для відображення тексту перекресленими символами.

Наприклад:

<s>Це перекреслені символи</s>

Тег <sub> – призначений для відображення тексту в вигляді верхнього індексу.

Наприклад:

_{Верхній індекс}

Тег <sup> – призначений для відображення тексту в вигляді нижнього індексу.

Наприклад:

^{Нижній індекс}

Теги <sub> та <sup> зручно використовувати для виводу формул.

Всі теги форматування тексту можуть бути вкладені один в інший. Це дозволяє проводити комбінований вивід тексту.

Наприклад, для того, щоб текст був виведений курсивом та був підкресленим необхідно:

<i><u>Приклад підкресленого курсиву</u></i>

Порядок вкладення тегів один в інший значення не має.

Наприклад, попередній приклад коректно записати і так:

<u><i>Приклад підкресленого курсиву</i></u>

Зміна зовнішнього вигляду тексту в вікні браузера при використанні наведених тегів форматування показана на рис. 1.2.

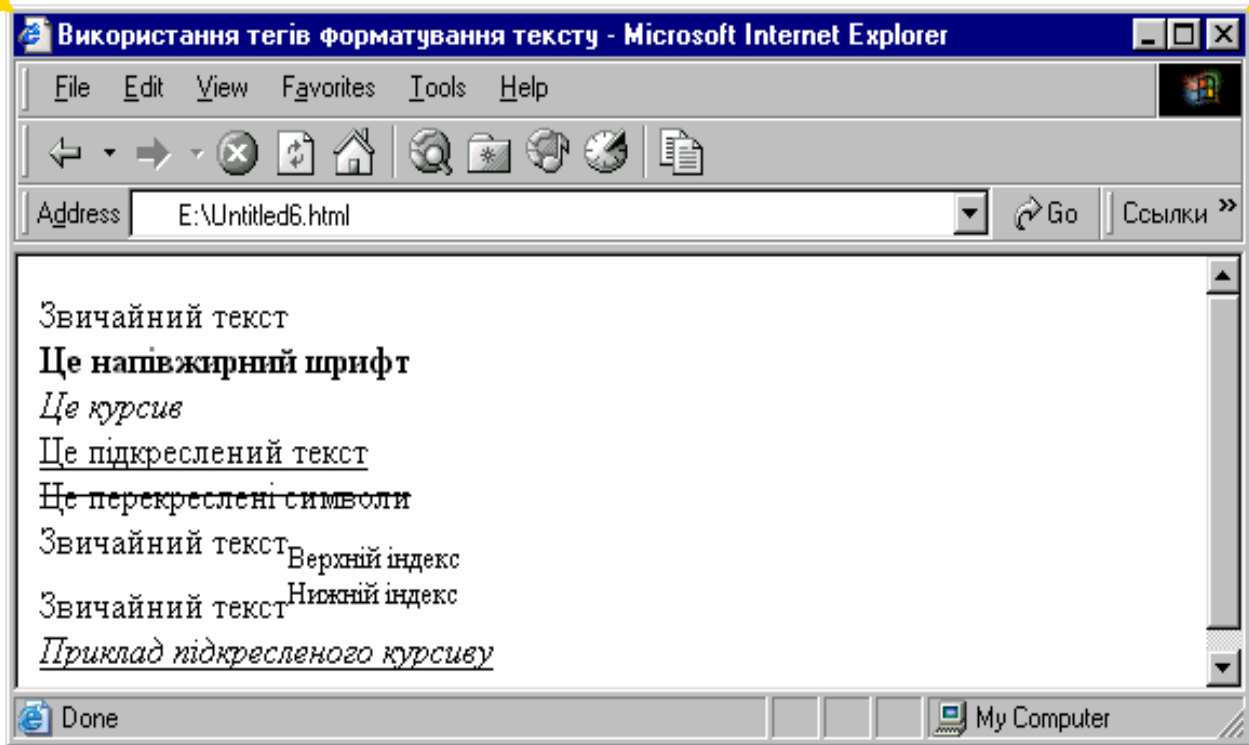
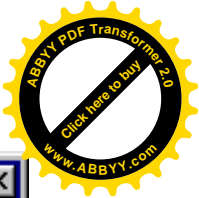


Рис. 1.2 - Використання тегів форматування тексту

Тег **** – призначений для визначення параметрів шрифту. За допомогою цього тегу можливо визначити колір символів, їх розмір та тип шрифту. Зміна параметрів шрифту відбувається за допомогою змін параметрів тегу **color**, **size** та **face**.

Параметр **color**. Дозволяє змінювати колір символів. Наприклад, визначення зеленого кольору символів можливо реалізувати так:

```
<font color="green">Цей текст зеленого кольору</font>
```

Відзначимо, що колір тексту, визначений за допомогою цього тегу, має пріоритет над кольором, визначеним за допомогою тегу **<body>**. Тобто, якщо в тегу **<body text="red">** визначено, що колір тексту червоний, а в тегу **** заданий жовтий колір, то на всій HTML-сторінці текст буде червоний, але блок, обмежений тегом ****, буде жовтого кольору.

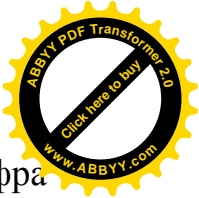
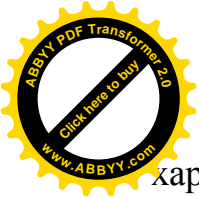
Параметр **face**. Дозволяє задати тип шрифту, яким браузер буде виводити текст. Для цього в якості значення параметру необхідно вказати назву шрифту. Наприклад, вивід тексту шрифтом Courier New, можливо реалізувати так:

```
<font face="Courier New"> Використовуємо шрифт Courier New</font>
```

Відзначимо, якщо на комп'ютері користувача такого шрифту немає, то браузер буде використовувати стандартний шрифт. Крім того можливо задати не одну, а декілька назв шрифтів, розділених комами. Якщо на комп'ютері користувача не буде першого шрифту, то браузер намагатиметься знайти шрифт за номером два і так далі. Наприклад:

```
<font face="Arial,Helvetica, sans-serif"> Використовуємо шрифт Arial, якщо такого нема, то Helvetica і так далі.</font>
```

Параметр **size**. Дозволяє визначити розмір символів. Розмір задається в умовних одиницях від 1 до 7. Фактичний розмір умовних одиниць залежить від



характеристик екрану комп'ютера користувача та настройок браузера. Цифра один відповідає найменшим символам, а 7 – найбільшим. Розмір символів при стандартних установках 3. Розмір символів можливо вказувати як в абсолютних, так і в відносних величинах.

Синтаксис запису наступний:

`Визначення символів в абсолютних величинах`

`Визначення символів в відносних величинах`

В першому випадку x – це цифра від 1 до 7, тобто величина символів в умовних одиницях. В другому випадку x може бути як додатнім, так і від'ємним числом. Причому знак $+$ або $-$ треба вказувати обов'язково. Величина x буде показувати на скільки умовних одиниць повинна змінитись величина символів відносно поточного значення.

Наприклад, вивід тексту з розміром символів 7 та 1 можливо реалізувати так:

`Визначення символів розміром 7`

`Визначення символів розміром 1`

Зменшення символів на одну умовну одиницю відносно поточного значення:

`Зменшення символів на 1 умовну одиницю `

Збільшення символів на одну умовну одиницю відносно поточного значення:

`Збільшення символів на 1 умовну одиницю `

Приклади використання параметрів `face` та `size` тегу `` показані на рис. 1.3.

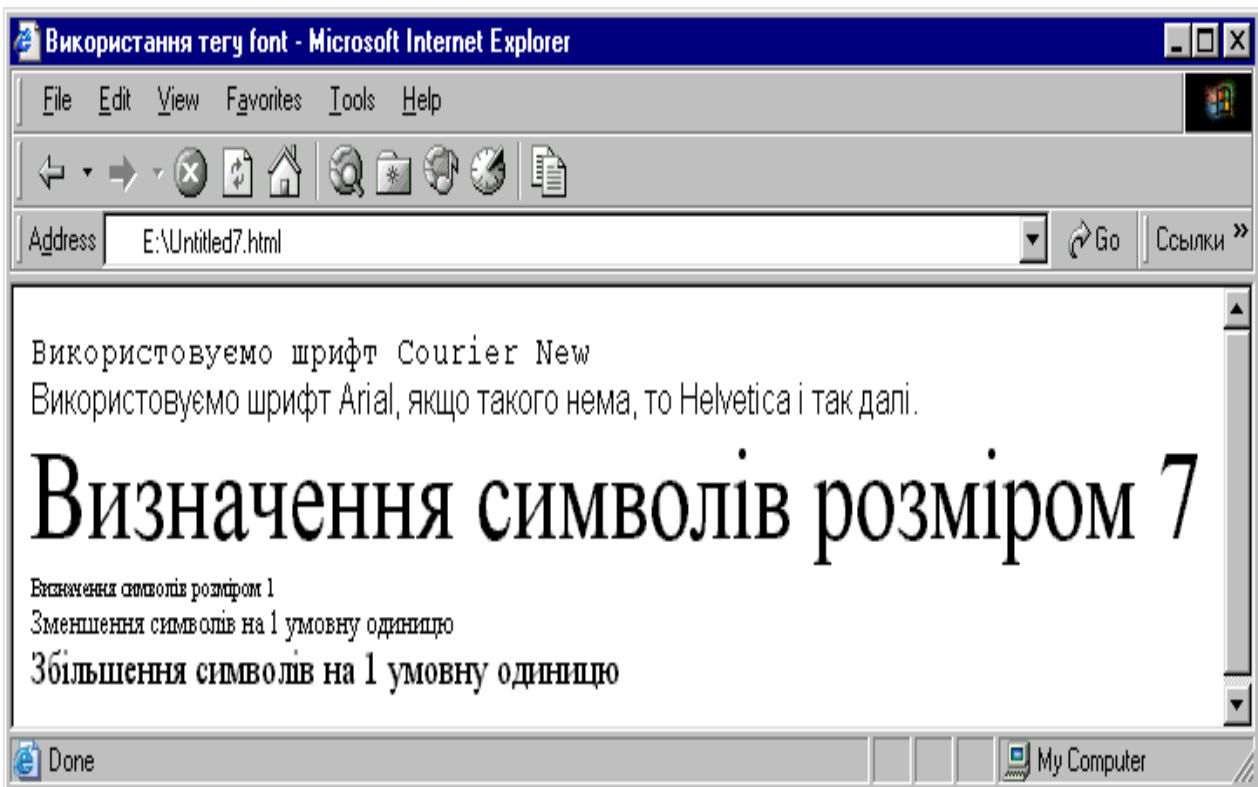
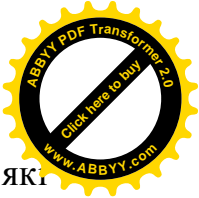


Рис. 1.3 Приклади використання параметрів `face` та `size` тегу ``



Тег **<basefont>** використовується для визначення параметрів шрифту, які будуть використовуватись в якості стандартних при перегляді HTML-сторінки. Параметри тегу **<basefont>** ті ж самі, що і для тегу ****. Зміна параметрів **<basefont>** розповсюджується на ту частину HTML-сторінки, яка розміщена нижче тегу **<basefont>**. При сумісному використанні тегу **<basefont>** з **** пріоритет має тег ****. На відміну від **** тег **<basefont>** не має закриваючого тегу. Його дія розповсюджується до кінця HTML-сторінки або до ще одного тегу **<basefont>**. Наприклад, визначення шрифту розміром 4 за допомогою тегу **<basefont>** можна здійснити так:

<basefont size=4>Використання тегу basefont

Відзначимо, що теги **** та **<basefont>** відносяться до послідовних тегів, а тому не можуть включати в себе теги рівня блоку, наприклад, **<table>** або **<tr>**.

1.4 Використання тегів заголовків

Теги заголовків (**<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>**, **<h6>**) призначені для виділення на екрані користувача певних фрагментів тексту HTML-сторінки. Виділення тексту реалізується за рахунок зміни розмірів та "жирності" тексту. Тегу **h1** відповідає найбільший розмір тексту, а **h6** – найменший розмір. Особливістю даних тегів є вставка порожнього рядка до і після виділеного фрагменту тексту.

Наприклад, для виділення в тексті HTML-сторінки трьох фрагментів тексту як заголовків 1, 3 та 6 рівнів необхідно:

<body>

Звичайний текст. <h1> Заголовок 1 рівня </h1>

Звичайний текст. <h3> Заголовок 3 рівня </h3>

Звичайний текст. <h6> Заголовок 6 рівня </h6>

Звичайний текст.

</body>

Відповідне відображення HTML-сторінки в вікні браузера показано на рис.

1.4.

Всі вказані теги є парними та можуть використовуватись з необов'язковим параметром **align**, який служить для горизонтального вирівнювання тексту. Можливі значення параметру **align** показані в табл. 1.2.

Таблиця 1.2 - Значення параметру align

Значення параметру	Наслідки використання
left	Текст вирівнюється по лівій стороні вікна браузера
center	Текст вирівнюється по центру вікна браузера
right	Текст вирівнюється по правій стороні вікна браузера
justify	Текст вирівнюється по ширині вікна браузера. Візуально відрізняється від left тільки в тому випадку, коли текст заголовку містить більш ніж два рядки тексту.

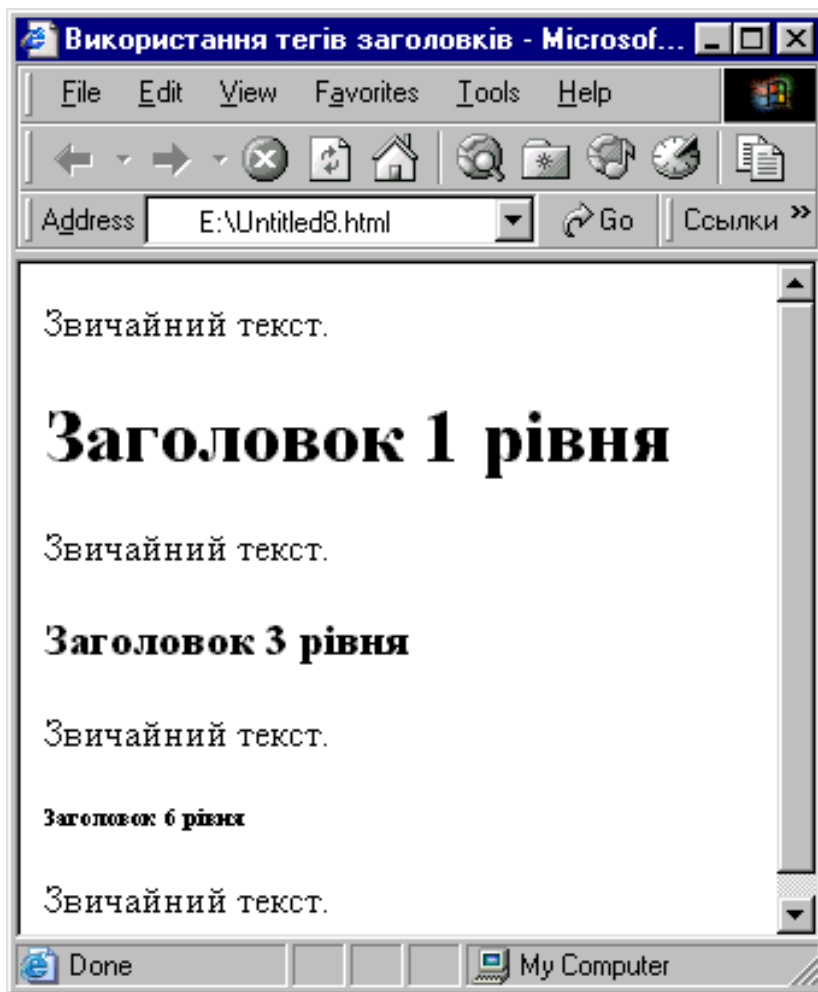


Рис. 1.4 - Використання тегів заголовків

Відзначимо, якщо параметр `align` не заданий, то використовується вирівнювання заголовку по центру вікна браузера.

Розглянемо приклад визначення на HTML-сторінці чотирьох заголовків третього рівня з різними параметрами вирівнювання. Для цього в розділі необхідно записати:

`<body>` Звичайний текст.

`<h3 align="center">` Заголовок вирівнюється по лівій стороні вікна браузера`</h3>`

Звичайний текст.

`<h3 align="center">` Заголовок вирівнюється по центру вікна браузера`</h3>`

Звичайний текст.

`<h3 align="right">` Заголовок вирівнюється по правій стороні вікна браузера`</h3>`

Звичайний текст.

`<h3 align="justify">` Заголовок вирівнюється по ширині вікна браузера.

Візуально відрізняється від `left` тільки в тому випадку, коли текст заголовку містить більш ніж два рядки тексту. `</h3>`

Звичайний текст. `</body>`

Відповідне відображення в вікні браузера показано на рис. 1.5.

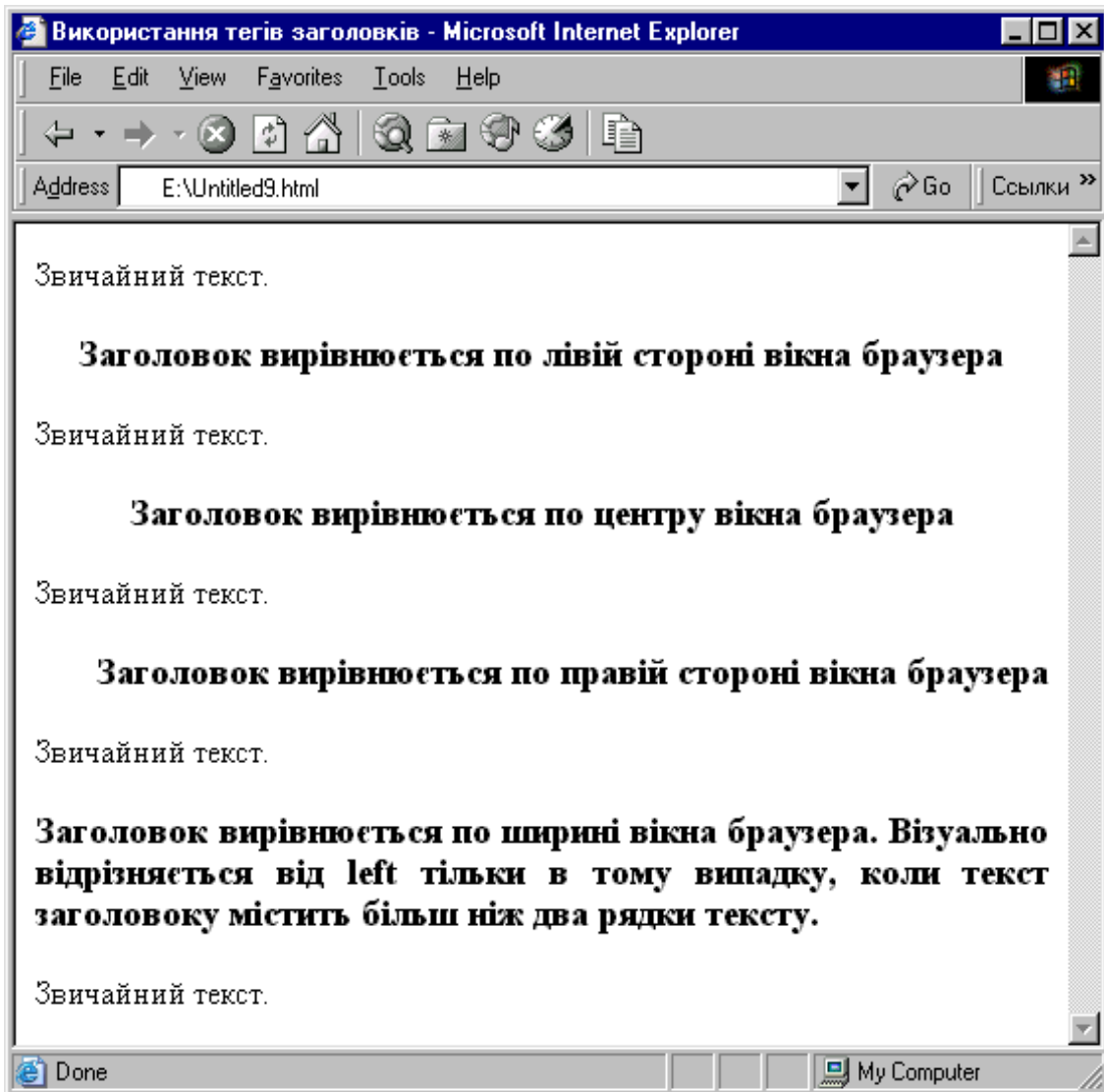


Рис. 1.5 - Використання тегу заголовку третього рівня з різними параметрами вирівнювання.

1.5 Форматування HTML-документу

Тег `
` використовується для примусового переведення рядка. Необхідність використання даного тегу пояснюється тим, що при відображенні текстових документів перенесення тексту із одного рядка в інший відбувається автоматично та залежить в основному від розмірів шрифту та розмірів вікна браузера. При цьому переведення рядка може здійснюватись тільки по символам, що розділяють окремі слова.

Розглянемо приклад запису виразів "Перший рядок" та "Другий рядок" в двох різних рядках. Відповідний HTML-код такий:

```
<body>  
Перший рядок <br> Другий рядок  
</body>
```


Парний тег `<nobr>` на відміну від тегу `
` використовується для заборони переводу рядка. Наприклад, для гарантованого розміщення в одному рядку виразу "Цей текст повинен відображатись в одному рядку" необхідно:

```
<nobr> Цей текст повинен відображатись в одному рядку </nobr>
```

Якщо даний текст не буде вміщуватись в одному рядку, то в вікні браузера з'явиться горизонтальна смуга прокрутки.

Тег `<p>` призначений для розділу тексту HTML-сторінки на окремі абзаци. Даний тег є тегом-контейнером та може використовуватись з необов'язковим параметром `align`. Значення параметру `align` та наслідки його використання для абзацив ті ж самі, що і для заголовків, див. табл. 2.2. Відзначимо, що візуально в вікні браузера абзац буде відділятися від іншого тексту за допомогою порожніх рядків та можливої зміни горизонтального вирівнювання. Наведемо приклад використання тегу `<p>`:

```
<body>
```

Звичайний текст

```
<p align="center"> Абзац з горизонтальним вирівнюванням - "по центру" </p> </body>
```

Відповідне вікно браузера показано на рис. 1.6.

Парний тег `<center>` призначений для горизонтального вирівнювання елементів HTML-сторінки посередні вікна браузера. Наведемо приклад його використання:

```
<body>
```

Звичайний текст

```
<center> Текст з горизонтальним вирівнюванням - "по центру" </center>
```

```
Звичайний текст </body>
```

Відзначимо, що в специфікаціях діючих стандартів HTML тег `<center>` позначений як не бажаний для використання, хоча і підтримується практично всіма розповсюдженими браузерами.

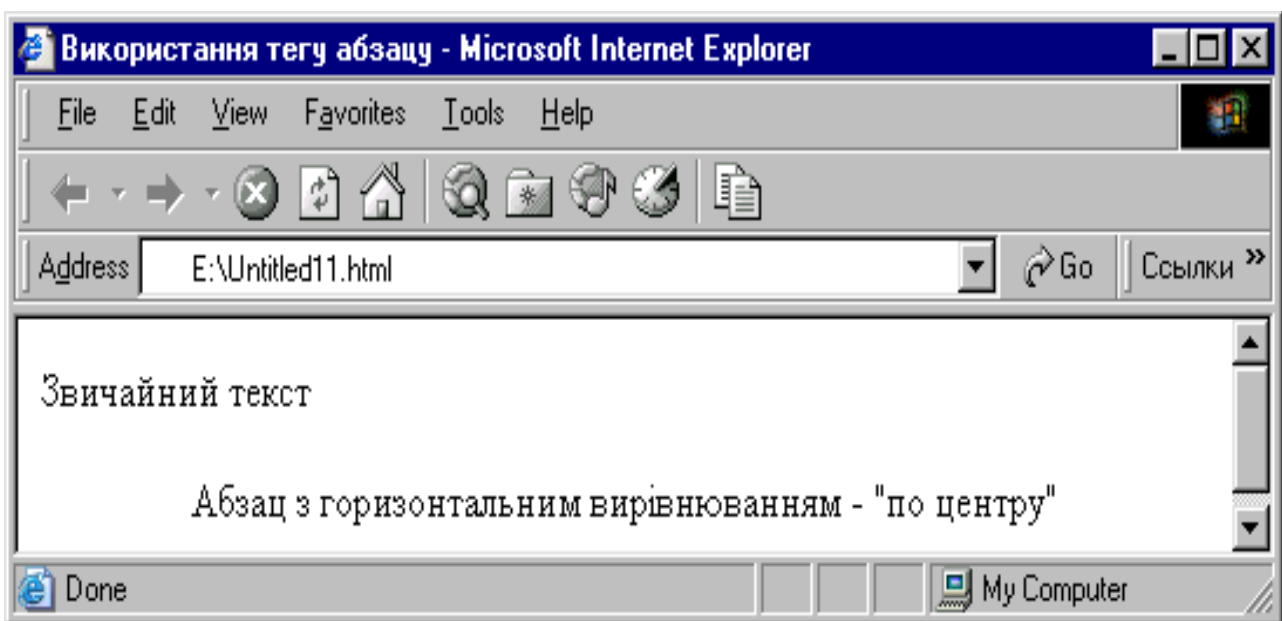
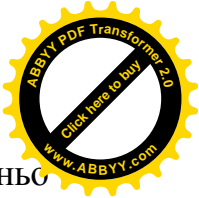


Рис. 1.6 - Використання тегу абзацу



Тег-контейнер `<pre>` використовується для визначення попередньо відформатованого тексту. Текст, розміщений між тегам `<pre>` та `</pre>`, буде виведений у вікні браузера із збереженням деяких параметрів форматування, виконаних за допомогою текстових редакторів. Відзначимо, що в середині контейнеру `<pre>` не можна використовувати деякі важливі та широко вживані теги. Тому використання цього тегу досить обмежене.

Тег `<hr>` призначений для виведення в вікні браузера горизонтальної лінії, до і після якої вставляються порожні рядки. Необов'язкові параметри тегу `<hr>` та їх призначення наведені в табл. 1.3.

Таблиця 1.3 - Параметри тегу `<hr>`

Параметр	Призначення
<code>align</code>	Визначення горизонтального вирівнювання лінії. Можливі значення: <code>left</code> ("по лівому краю"), <code>center</code> ("по центру"), <code>right</code> ("по правому краю"). Якщо параметр <code>align</code> не використаний то застосовується вирівнювання "по центру"
<code>color</code>	Встановлює колір лінії
<code>noshed</code>	Відмінняє рельєфність лінії
<code>size</code>	Визначає товщину лінії
<code>width</code>	Встановлює довжину лінії

Наприклад, для визначення лінії червоного кольору, товщиною 2 пікселі, довжиною 400 пікселів, без рельєфу необхідно:

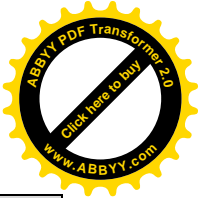
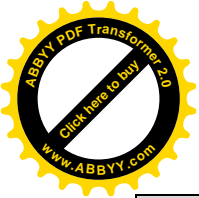
```
<hr color="red" size=2 width=400 noshade>
```

1.6 Використання коментарів та спеціальних символів

В деяких випадках необхідно не показувати у вікні браузера частину HTML-коду сторінки, тобто "за коментувати" цей код. Для цього використовуються теги коментарів `<!--` та `-->`. Наприклад:

```
<!-- Цей текст не буде відображатись у вікні браузера -->
```

Як відомо, при створенні HTML-сторінки повинні застосовуватись тільки символи, що входять до базової частини таблиці кодів ASCII. Проте деякі спеціальні символи (наприклад, пробіл) не входять до цієї частини таблиці. В випадку використання вказаних спеціальних символів необхідно застосовувати спеціальні HTML-коди, див. табл. 1.4.



Таблиця 1.4 - Коды спеціальних символів

Спеціальний символ	HTML-код
Амперсанд (&)	&
Знак copyright (©)	©
Знак більше (>)	>
Знак менше (<)	<
Пробіл	
Знак плюс/мінус (±)	±
Знак одна друга (½)	½
Знак одна друга (¼)	¼
Знак одна друга (¾)	¾
Знак "»"	»
Знак "§"	§
Знак "«"	«

1.7 Рисунки

Для вставки рисунка в HTML-сторінку використовується тег ``. Основним параметром цього тегу є параметр *src*, за допомогою якого вказується адреса файлу, в якому і зберігається рисунок. Цей же тег також можливо використовувати для показу відео-роликів. В цьому випадку використовується параметр *dynsrc*, за допомогою якого вказується адреса відео файлу. Найпоширенішими форматами файлів-рисунків є формати *jpg*, *gif*, *png*, а відео файлів - *avi*.

Відзначимо, що адреса файлу-рисунок може бути вказана в абсолютному та відносному форматі. При використанні абсолютного формату в адресі повністю визначено комп'ютер, каталог та файл рисунок.

Наприклад, для вставки рисунка, що зберігається в файлі *logo.gif* за адресою www.picture.com.ua/picture необхідно:

```

```

При використанні відносного формату визначення місцезнаходження файлу-рисунок відбувається з урахуванням місцезнаходження даної HTML-сторінки.

Наприклад, для вставки в HTML-сторінку *prim.html* рисунка, який зберігається в файлі *logo.gif*, що знаходиться в одному каталозі з файлом *prim.html* необхідно:

```

```

Графічні файли зручно зберігати в окремому каталозі, наприклад, в каталозі з іменем *img*. В цьому випадку для вставки в HTML-сторінку *prim.html* рисунка *logo.gif* необхідно:

```

```

Відзначимо, що каталог *img* та файл *prim.html* повинні знаходитись в одному каталозі. Крім обов'язкового параметру *src* тег `` має декілька необов'язкових параметрів, див. табл. 1.5.

Таблиця 1.5 - Параметри тегу

Назва параметру	Призначення
alt	Визначення альтернативного тексту
width	Ширину рисунку в пікселях
height	Висоту рисунку в пікселях
hspace	Відступ від рисунку до інших об'єктів на HTML-сторінці по вертикалі
vspace	Відступ від рисунку до інших об'єктів на HTML-сторінці по горизонталі
border	Товщина рамки навколо рисунку
lowsrc	Задає файл з альтернативним зображенням
align	Вирівнювання рисунку відносно інших об'єктів на HTML-сторінці
src	Адреса графічного файлу
dynsrc	Адреса відео файлу
start	Задає момент початку прокрутки відео ролика. Можливі значення: fileopen – ролик починається відразу після завантаження відео файлу, mouseover – ролик починається при наведенні миші на зображення.
loop	Задає кількість повторів відео - ролика. Якщо значення дорівнює -1, то програвання відбувається безперервно

Розглянемо призначення вказаних параметрів докладніше. Досить часто при перегляді HTML-сторінок користувачі використовують браузер, що працює в режимі відключення завантаження зображень. Як відомо, цей режим дозволяє зменшити час відображення HTML-сторінки в вікні браузера. В цьому випадку замість зображення на екрані з'являється альтернативний текст, заданий в параметрі *alt*. Крім цього альтернативний текст відображається в якості підказки при знаходженні курсору "миші" в області рисунку. Приклад використання параметру *alt*:

```

```

Параметри *width* та *height* призначені для визначення розмірів рисунку у вікні браузера. Відзначимо, що при зміні розмірів зображення розмір файлу не змінюється, тобто час завантаження рисунку не змінюється. Наведемо приклад застосування параметрів *alt*, *width* та *height* для одного і того рисунку:

```

```

```

```

```

```

Відповідне вікно браузера показано на рис. 1.7.

Слід зазначити, що зміна розмірів рисунку може призвести до втрати якості зображення, особливо за рахунок втрати пропорцій між початковою висотою та шириною. Практичний досвід показує, що навіть при дотриманні пропорцій збільшення початкових розмірів зображень-фотографій більш ніж на 10-15% призводить до чуттєвої втрати якості. При цьому зменшення розмірів зображень позначається на якості їх відображення набагато менше, але їх використання не оптимальне з точки зору часу завантаження.

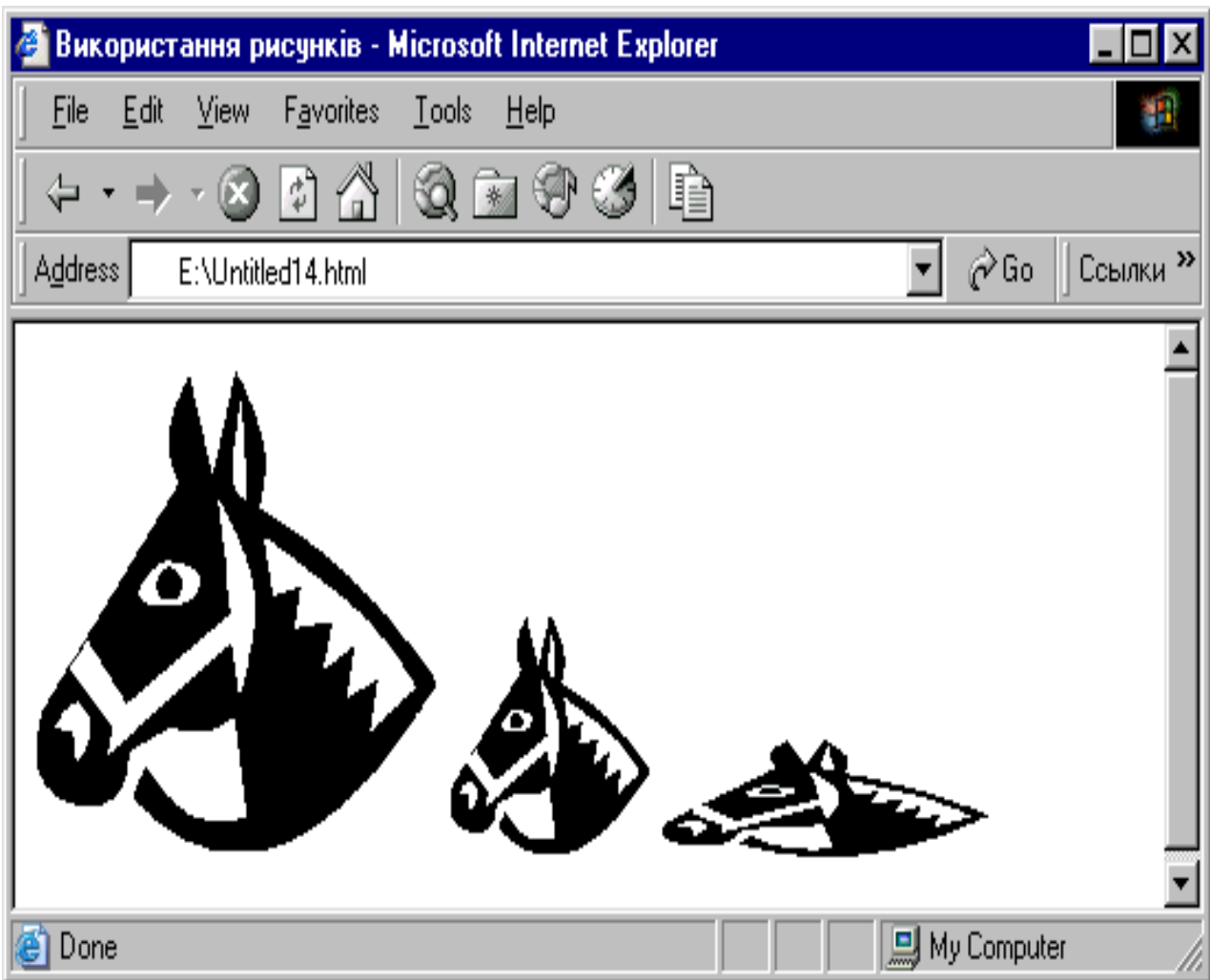


Рис. 1.7 - Використання параметрів alt, width та height тегу

Параметр **border** дозволяє визначити товщину рамки навколо рисунку. Застосуємо цей параметр із значенням 2 для рисунків попереднього прикладу:

```

```

```

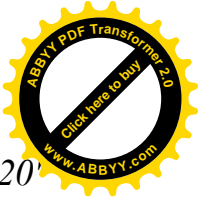
```

```

```

Параметри **hspace** та **vspace** дозволяють в пікселях задавати відступи від рисунку до інших об'єктів на HTML-сторінці.

Покажемо застосування цих параметрів із значеннями 20 та 20 для попереднього прикладу:



```

```

```

```

```

```

Параметр **lowsrc** дозволяє визначити файл з альтернативним зображенням, яке з'являється у вікні браузера до появи основного зображення. Альтернативне зображення, як правило, це рисунок того ж змісту, що і основний, але менш якісний. По цій причині файл альтернативного зображення менший за розміром, а тому завантажується набагато швидше ніж основний. Це дозволяє швидко показати загальні риси зображення.

Приклад використання:

```

```

Параметр **align** використовується для визначення горизонтального або вертикального вирівнювання рисунку відносно інших об'єктів на HTML-сторінці. Параметри вирівнювання залежать від значень параметру align. При використанні цього параметру із значеннями **left** або **right** рисунок розміщується відповідно до лівої або правої сторони вікна браузера, а текст "обтікає" зображення з протилежної сторони. Відзначимо, що в цьому випадку текст може займати декілька рядків.

Приклад використання параметру align:

```
<body>
```

```

```

Приклад використання параметру align="right". Рисунок розміщений біля правої сторони вікна браузера. Текст обтікає зображення з лівої сторони.

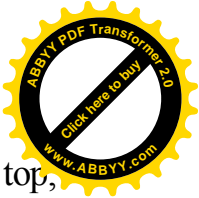
```
</body>
```

Крім **left** та **right** параметр **aling** може приймати наступні значення:

- **top** – верхня межа зображення вирівнюється по найбільш високому елементу рядка;
- **texttop** – верхня межа зображення вирівнюється по найбільш високому текстовому елементу рядка;
- **middle** – середина зображення вирівнюється по базовій лінії рядка;
- **absmiddle** – середина зображення вирівнюється по середині рядка;
- **bottom** – нижня межа зображення вирівнюється по базовій лінії рядка;
- **absbottom** – нижня межа зображення вирівнюється по нижній межі рядка.

В цих випадках зображення можна розглядати як звичайний елемент рядка.

Відзначимо, що базова лінія тексту – це нижня лінія рядка тексту без врахування нижньої частини деяких символів (p, j, y).



Наведемо приклад використання параметру align тегу з значеннями top, middle та bottom:

```
<body>
```

```

```

Приклад використання параметру align="top".


```

```

```
</body>
```

1.8 Звук

Для програвання звуку на Web-сторінках можливо використовувати тег <embed>, основним параметром якого є *src*, що дозволяє визначити адресу звукового файлу. Наприклад, для того, щоб при перегляді в браузері Web-сторінки прозвучав звук, записаний в файлі *1.wav*, необхідно записати такий HTML-код: <embed src=1.wav >. Основні параметри цього тегу показані в табл. 1.6.

Таблиця 1.6 - Парметри тегу <embed>

Назва параметру	Призначення
width	Ширину звукового програвача в пікселях
height	Висоту звукового програвача в пікселях
hspace	Відступ від звукового програвача до інших об'єктів на HTML-сторінці по вертикалі
vspace	Відступ від звукового програвача до інших об'єктів на HTML-сторінці по горизонталі
hidden	Дозволяє показувати/не показувати (false/true) звуковий програвач у вікні браузера
lowsrc	Задає файл з альтернативним зображенням
align	Вирівнювання рисунку відносно інших об'єктів на HTML-сторінці
src	Адреса графічного файлу
dynsrc	Адреса відео файлу
autostart	Задає момент початку прокрутки відео ролика. Можливі значення: true – програвання починається відразу після завантаження звукового файлу, false – програвання починається після команди користувача.
loop	Задає кількість повторів відео - ролика. Можливі значення: true – програвання відбувається безперервно, false – кількість повторів визначається користувачем.



1.9 Гіперпосилання

Як відомо, гіперпосилання призначені для зв'язування одного Web-ресурсу з іншим. Гіперпосилання складається із двох частин. Перша частина – це об'єкт, який користувач бачить у вікні браузера, і вибір якого призводить до переходу на цільовий Web-ресурс, URL-адреса якого вказана в другій частині гіперпосилання. В якості цього об'єкту можуть бути текст або/та зображення. В залежності від цього класифікують текстові та графічні гіперпосилання. При типових настройках браузера гіперпосилання виділяються на HTML-сторінці за допомогою кольору. Крім того текст гіперпосилання виділений за допомогою підкреслення. Для визначення гіперпосилання в коді HTML-сторінки використовують парний тег `<a>`. Основним параметром цього тегу є параметр *href*, що задає URL-адресу гіперпосилання. Синтаксис запису гіперпосилання такий:

```
<a href="URL-адресу"> Текстовий або графічний об'єкт </a>
```

Приклади запису текстового та графічного гіперпосилання показані нижче:

```
<a href="www.meta.ua"> Текстове гіперпосилання </a>
```

```
<a href="www.meta.ua">  </a>
```

Вказана URL-адреса може бути відносна або абсолютна. При використанні відносної адреси повний шлях до цільового ресурсу не заданий. В цьому випадку визначення місцезнаходження цільового ресурсу відбувається з урахуванням місцезнаходження HTML-сторінки, в якій використане гіперпосилання. Наприклад, якщо в файлі 1.html визначене гіперпосилання:

```
<a href="2.html"> Текстовий або графічний об'єкт </a>
```

то мається на увазі, що цільова HTML-сторінка 2.html знаходиться в тому ж каталозі того ж комп'ютера, в якому знаходиться і HTML-сторінка 1.html.

Абсолютною називається URL-адреса, в якій повністю визначено комп'ютер, каталог та файл цільового ресурсу. Наприклад:

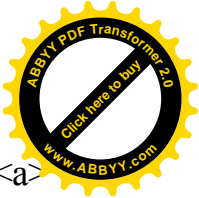
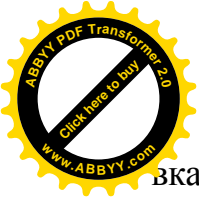
```
<a href="www.example.com/html/3.html"> Абсолютне гіперпосилання </a>
```

Важливим параметром тегу `<a>` є параметр *target*, за допомогою якого можливо вказати вікно браузеру, в яке повинен бути завантажений цільовий ресурс. Для завантаження цільового ресурсу в нове вікно браузера значення цього параметру має бути "*blank*". Наприклад, для визначення текстового гіперпосилання, вибір якого призводить до відображення в новому вікні браузера файлу 2.html, необхідно:

```
<a href="2.html" target="_blank"> Текст </a>
```

Відзначимо, що при відсутності параметру *target* цільовий ресурс буде завантажуватись в вікно браузера, в якому відображається HTML-сторінка з гіперпосиланням.

Цікавою є можливість гіперпосилань задати в якості цільового ресурсу певну частину власної HTML-сторінки. Наприклад, зручно читати великий документ, коли він має зміст з гіперпосиланнями на різні розділи. В цьому випадку перехід по гіперпосиланню до деякого розділу (який може знаходитись в кінці документу) призведе до відображення цього розділу на екрані. Такі гіперпосилання називаються внутрішніми. Для їх побудови необхідно за допомогою тегу `<a>`



вказати місце переходу (покажчик) та за допомогою параметру *name* тега `<a>` присвоїти йому ім'я.

Наприклад, для переходу на 12 розділ документу на початку тексту цього розділу розмістимо покажчик з іменем "go12":

```
<a name="go12"></a>Текст розділу 12
```

Відзначимо, що в цьому випадку параметр `href` не використовується, а ім'я покажчика повинно бути записане за допомогою літер англійського алфавіту та цифр.

Особливістю гіперпосилання для переходу в середині HTML-сторінки є використання в параметрі `href` імені покажчика з перфіксом `#`. Наприклад:

```
<a href="#go12">Перехід до розділу 12</a>
```

1.10 Списки

Списки – широко розповсюджена форма показу даних як в електронних так і в друкованих документах. Мовою HTML передбачено використання трьох стандартних видів списків: маркірованого, нумерованого та списку визначень. Відзначимо, що від інших елементів на HTML-сторінці стандартні списки відділяються порожніми рядками.

1.10.1 Маркіровані списки

Для визначення маркірованого списку використовується тег-контейнер ``, в середині якого розміщуються всі елементи списку. При цьому кожен пункт списку повинен починатись тегом ``. Наведемо приклад HTML-коду маркірованого списку:

```
<body> Звичайний текст  
<ul>  
<li> Перший пункт маркірованого списку  
<li> Другий пункт маркірованого списку  
<li> Третій пункт маркірованого списку  
</ul>  
Звичайний текст  
</body>
```

Відображення маркірованого списку у вікні браузера показано на рис. 1.8.

Як в тезі ``, так і в тезі `` можливо використовувати необов'язковий параметр *type*, за допомогою якого визначається тип маркера списку. Можливі значення цього параметру:

- *disc* – маркери відображаються заповненими колами;
- *circle* – маркери відображаються не заповненими колами;
- *square* – маркери відображаються заповненими квадратами.

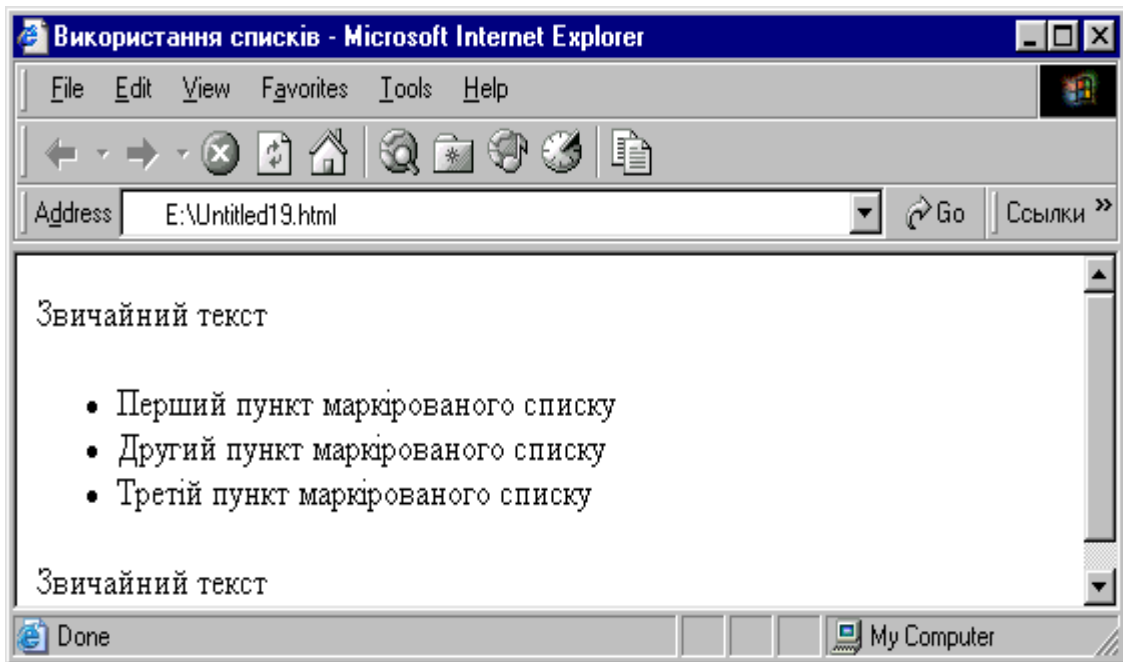


Рис. 1.8 - Відображення маркірованого списку

Приклад використання різних значень параметру `type` в тезі `` маркірованого списку:

```
<ul>
<li type="disc"> Перший пункт маркірованого списку. type="disc"
<li type="square"> Другий пункт маркірованого списку.
type="square"
<li type="circle"> Третій пункт маркірованого списку. type="circle"
</ul>
```

Відповідне відображення у вікні браузера показано на рис. 1.9.

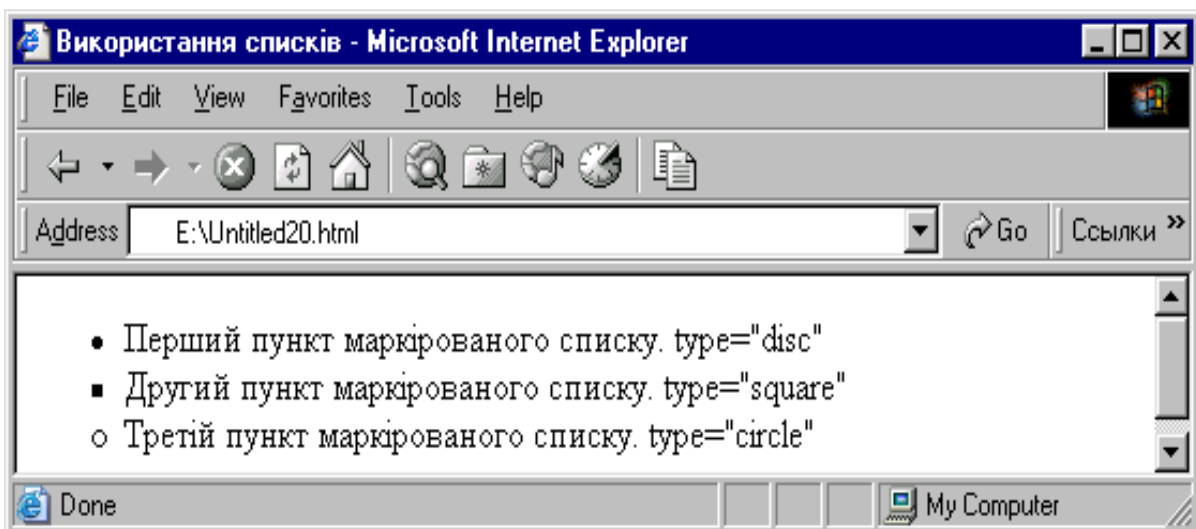


Рис. 1.9 - Використання різних значень параметру `type` в тезі ``

Якщо параметр `type` в тегах `` та `` не використовується, то для всього списку використовується маркер типу "disc".

1.10.2 Нумеровані списки

Для визначення нумерованого списку використовується тег-контейнер ``, в середині якого розміщуються всі елементи списку. При цьому кожен пункт списку повинен починатись тегом ``. Наведемо приклад HTML-коду нумерованого списку:

```
<ol>
<li> Перший пункт нумерованого списку
<li> Другий пункт нумерованого списку
<li> Третій пункт нумерованого списку
</ol>
```

Відображення такого нумерованого списку у вікні браузера показано на рис. 1.10.

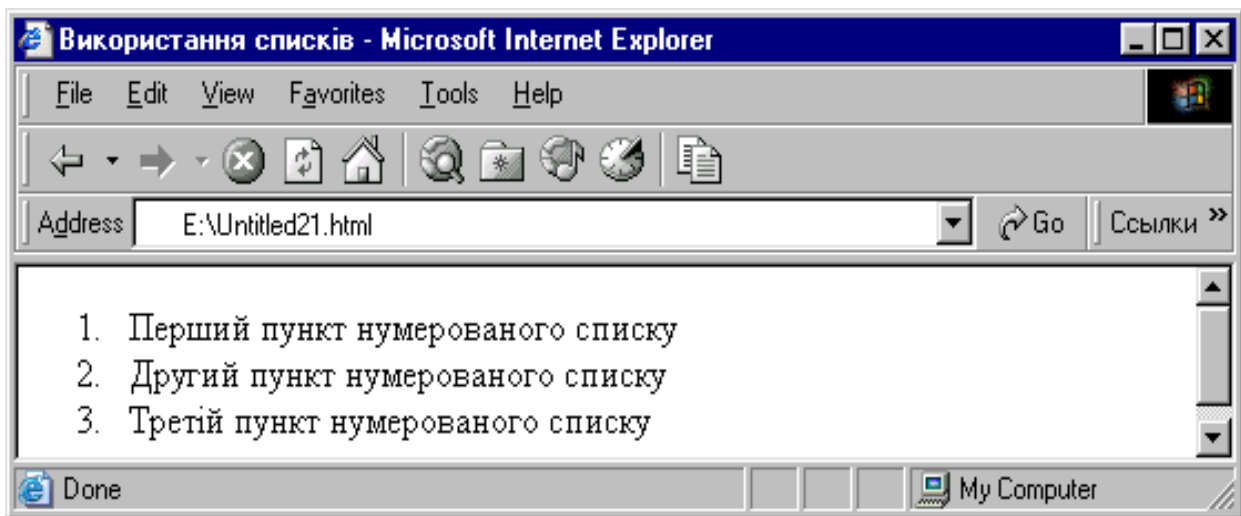


Рис. 1.10 - Відображення нумерованого списку

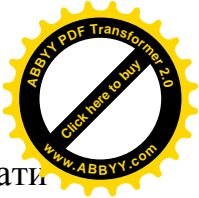
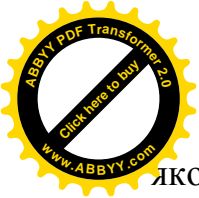
Досить часто тег `` використовується з необов'язковими параметрами *type* та *start*. При цьому параметр *type* використовується також в тезі ``. Його призначенням є визначення одного з стандартних типів маркера. Відзначимо, що для нумерованого списку існує п'ять стандартних типів маркерів. Використання цього параметру в тезі `` розповсюджується на весь список, а використання в тезі `` розповсюджується тільки на поточний пункт списку.

Відзначимо, що нумерація пунктів нумерованого списку не залежить від типу маркеру. Якщо параметр *type* в тегах `` та `` не зазначено, то для всього списку використовується маркер типу "1".

Параметр *start* призначений для зміни початку нумерації пунктів списку і може використовуватись тільки в тезі ``. Для зміни нумерації пунктів в середині списку в тезі `` використовується параметр *value*. Значеннями параметрів *start* та *value* можуть бути тільки натуральні числа.

1.10.3 Списки визначень

Список визначень використовується для розміщення на HTML-сторінці тексту, за змістом подібного до енциклопедії або словника. Тобто тексту, в



якому потрібно навести термін та пояснення цього терміну, що може займати декілька рядків. На відміну від інших типів списків кожен пункт списку визначень складається із двох частин.

В першій частині пункту записується термін, що потребує визначення.

В другій частині пункту записується текст, що пояснює зміст терміну.

Список визначень задається за допомогою тегу-контейнеру `<dl>`. Для визначення терміну використовується тег `<dt>`, а для визначення пояснення використовується тег `<dd>`.

Наведемо приклад списку визначень для розміщення на HTML-сторінці деяких стандартних термін в галузі Web-програмування:

```
<dl>
```

```
<dt>Інтернет (мережа Інтернет)
```

```
<dd>Сукупність мереж та обчислювальних засобів, які використовують стек протоколів TCP/IP (Transport Control Protocol/Internet Protocol), спільний простір імен та адрес для забезпечення доступу до інформаційних ресурсів мережі будь-якій особі;
```

```
<dt>Гіпертекстове посилання
```

```
<dd>Адреса іншого мережевого інформаційного ресурсу у форматі URL, який тематично, логічно або будь-яким іншим способом пов'язаний з документом, у якому це посилання визначене.
```

```
</dl>
```

Відображення цього списку визначень у вікні браузера показано на рис. 1.11.

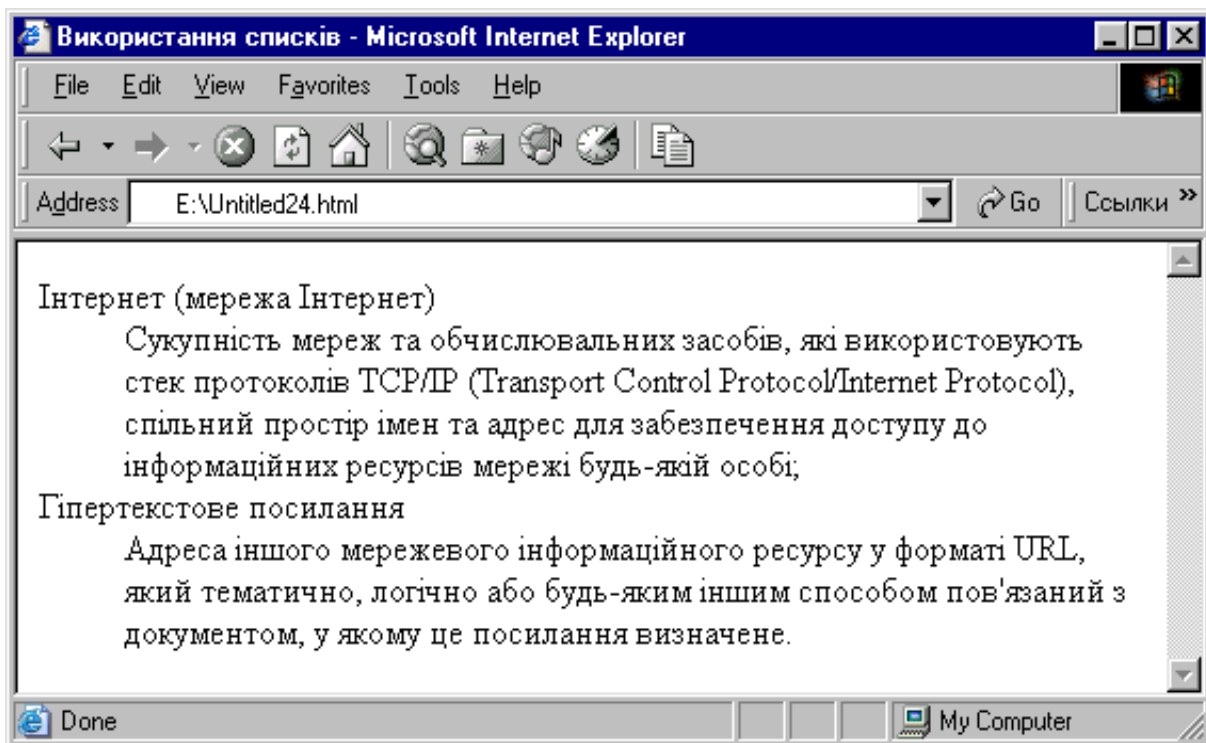


Рис. 1.11 - Список визначень



1.11 Таблиці

Особливість використання таблиць полягає в тому, що вони є основним засобом форматування HTML-документів. Розміщення інформації в HTML-документі, виконане за допомогою інших засобів (тегів), має або набагато менші можливості, або не однаково відображається браузерами різних типів.

Для побудови таблиці необхідно використати теги-контейнери: тег таблиці `<table>`, тег рядка `<tr>` та тег комірки даних `<td>` або тег комірки заголовка `<th>`. При цьому таблиця має бути розміщена в тілі HTML-документу, будується вона рядками, причому кількість комірок в рядках повинна бути однаковою. Таким чином одній таблиці відповідає один тег `<table>` та стільки тегів `<tr>`, скільки рядків має таблиця. Якщо в таблиці немає об'єднання комірок, то сумарна кількість тегів `<td>` та `<th>` дорівнює добутку кількості рядків на кількість колонок таблиці. Різниця між тегами `<td>` та `<th>` полягає в форматуванні та вирівнюванні розміщеного в них тексту. В тезі `<td>` текст відображається стандартним шрифтом з горизонтальним вирівнюванням "по лівому краю". В тезі `<th>` текст відображається напівжирним шрифтом з горизонтальним вирівнюванням "по середині". Для розміщення інформації (тексту або графіки) в середині таблиці необхідно помістити її в тег комірки. Таблиця може мати заголовок, якому відповідає парний тег `<caption>`, він має бути розміщений після тегу `<table>` до першого тегу `<tr>`.

Наведемо приклад HTML-коду для визначення таблиці, яка складається із заголовку та трьох рядків, в кожному із яких має бути дві комірки:

```
<table>
<caption>Заголовок таблиці </caption>
<tr><th>Рядок 1. Комірка 1</th><th>Рядок 1. Комірка 2</th></tr>
<tr><td>Рядок 2. Комірка 1</td><td>Рядок 2. Комірка 2</td></tr>
<tr><td>Рядок 3. Комірка 1</td><td>Рядок 3. Комірка 2</td></tr>
</table>
```

Для зручності ідентифікації комірок в них розміщено текст, що відповідає номеру рядка та номеру комірки в рядку. Відображення такої таблиці у вікні браузера показано на рис. 1.12.

При відсутності параметру *border* у тегах, що визначають таблицю, її межі не відображаються.

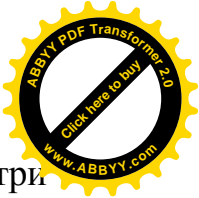
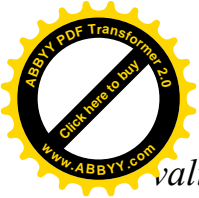
Розглянемо призначення найбільш використовуваних параметрів тегів `<table>`, `<caption>`, `<tr>`, `<td>` та `<th>`.

Тег `<caption>` найчастіше використовується тільки з параметрами **align** та **valign**, що визначають відповідно горизонтальне та вертикальне вирівнювання заголовка таблиці.

Можливі значення параметру *align* тегу `<caption>`:

- **right** - заголовок вирівнюється по правому краю таблиці;
- **center** - заголовок вирівнюється по центру таблиці;
- **left** - заголовок вирівнюється по лівому краю таблиці.

Параметр *valign* дозволяє розміщувати заголовок над таблицею (значення **top**) або під таблицею (значення **bottom**). При відсутності параметрів *align* та



valign вирівняний по центру заголовок розміщується над таблицею. Параметри тегу `<table>` та їх призначення наведені в табл. 1.7.

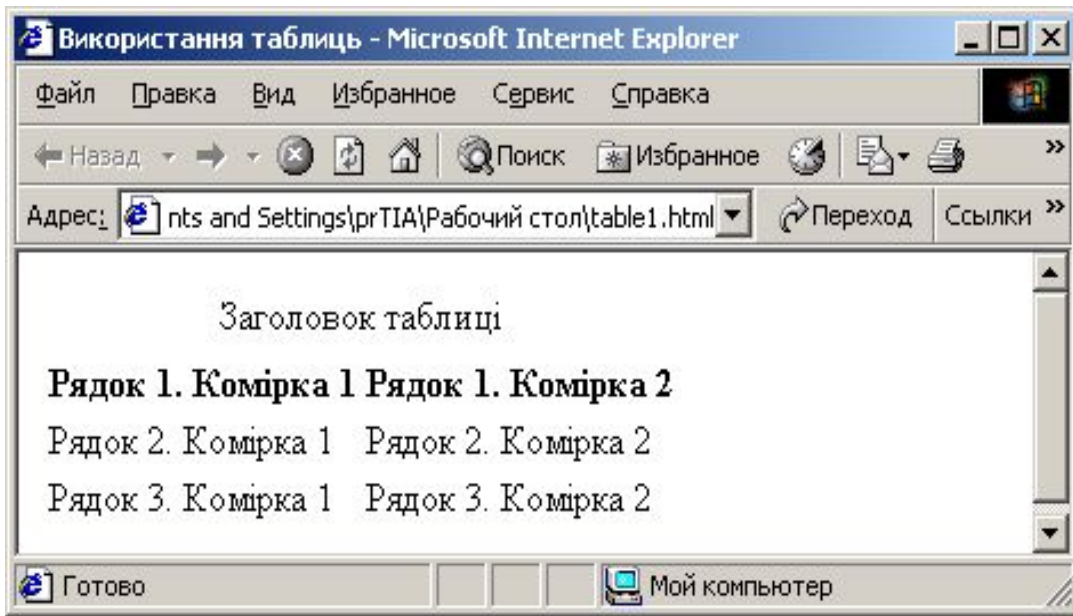


Рис. 1.12 - Використання таблиць

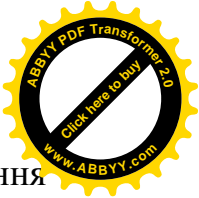
Таблиця 1.7 - Параметри тегу `<table>`

Параметр	Призначення
<i>border</i>	Визначає товщину межі таблиці в пікселях
<i>bordercolor</i>	Визначає колір меж таблиці
<i>cellspacing</i>	Визначає відстань між комірками таблиці в пікселях
<i>cellpadding</i>	Визначає відстань від меж комірок до даних, що знаходяться в цих комірках
<i>width</i>	Визначає ширину таблиці в пікселях або процентах від ширини HTML-документу
<i>height</i>	Визначає висоту таблиці в пікселях
<i>align</i>	Визначає горизонтальне вирівнювання таблиці
<i>bgsolor</i>	Визначає колір фону таблиці
<i>background</i>	Визначає фоновий рисунок таблиці

1.12 Фрейми

Фреймова структура HTML-документу дозволяє розділити вікно браузера на декілька прямокутних областей (**фреймів**), в кожену із яких можливо завантажити окремий HTML-документ.

Між фреймами можливо організувати **взаємодію**. Сенс **взаємодії між фреймами** полягає в тому, що вибір посилання в одному із фреймів може призвести до завантаження вказаного HTML-документу в інший фрейм.



В основному **фрейми застосовуються** для організації управління завантаженням HTML-документів в певну область вікна браузера при роботі користувача в іншій області, тобто для організації меню.

Хоча на сьогодні **фреймова структура** досить широко застосовуються, але вона **не відповідає сучасним тенденціям** в галузі побудови Web-сайтів.

Особливістю HTML-документу із фреймовою структурою є те, що кількість html-файлів на одиницю більше від кількості фреймів. Причиною цього є необхідність використання окремого html-файлу для визначення фреймової структури.

Фреймова структура HTML-документу визначається за допомогою тегу-контейнеру `<frameset>`, що використовується замість тегу `<body>`.

Тег `<frameset>` може мати параметри *rows* та *cols*, за допомогою яких визначається кількість та розміри областей, на які розділяється вікно браузера. За допомогою параметру *rows* вікно браузеру розділяється на горизонтальні області, а за допомогою параметру *cols* – на вертикальні області.

Синтаксис запису тегу:

```
<frameset rows="висота_1, висота_2..." cols="ширина_1, ширина_1...">
```

Кількість заданих "висот" та "ширин" задають кількість горизонтальних та вертикальних областей (**фреймів**), на які буде розділене вікно браузера. Для коректного визначення фреймової структури необхідно, як мінімум, задати хоча б два розміри для одного із параметрів *rows* чи *cols*.

Розміри фреймів можуть задаватись в **пікселях, процентах від розмірів вікна браузера** або **відносних одиницях**.

Наведемо приклади.

Визначення розмірів фреймів в пікселях:

`<frameset rows="100,500">` - визначає два горизонтальні фрейми висотою 100 та 500 пікселів.

Визначення розмірів фреймів в процентах:

`<frameset rows="10%,90%">` - визначає два горизонтальні фрейми висотою 10% та 90% від висоти вікна браузера.

Визначення розмірів фреймів в відносних одиницях:

`<frameset rows="*,2*">` - визначає два горизонтальні фрейми висотою $\frac{1}{3}$ та $\frac{2}{3}$ від висоти вікна браузера. В цьому випадку для показу відносних величин використовується символ *.

В середині контейнеру `<frameset>` можуть розміщуватись тільки теги визначення окремого фрейма `<frame>` або вкладені теги `<frameset>`. Сумарна кількість цих тегів повинна бути рівною кількості фреймів, заданих параметрами *rows* та *cols* контейнеру `<frameset>`.

Параметри тегу `<frame>` представлені в табл. 1.8.

Наведемо приклад html-коду для визначення двох фреймів, що займають відповідно 25% та 75% від ширини вікна браузера.

Відзначимо, що межа між фреймами повинна відображатись у вікні браузера.

Лівий фрейм має ім'я *fmenu*, в ньому завантажений файл *fmenu.html*.

Правий фрейм має ім'я *fwindow*, в ньому завантажений файл *fwindow.html*. Поля лівого фрейму становлять 5 пікселів, а поля правого фрейму 7 пікселів.

Розміри правого фрейму не можуть бути змінені користувачем.

```
<frameset cols="25%,75%">
    <frame src="fmenu.html" name="fmenu" frameborder="1"
marginwidth="5" marginheight="5">
    <frame src="fwindow.html" name="fwindow" marginwidth="7"
marginheight="7" noresize>
</frameset>
```

Відповідне вікно браузеру показано на рис. 1.13.

Таблиця 1.8 - Параметри тегу <frame>

Параметр	Призначення
<i>src</i>	Визначає URL- адресу документу, завантаженого в фрейм
<i>name</i>	Визначає ім'я фрейму, яке використовується для взаємодії між фреймами.
<i>frameborder</i>	Вказує на наявність ("1"), або відсутність ("0") межі між фреймами
<i>bordercolor</i>	Визначає колір межі (при <i>frameborder</i> ="1") між фреймами
<i>scrolling</i>	Задає наявність смуг прокрутки. "No" – немає, "Yes" – є, "Auto" – з'являються при необхідності
<i>marginwidth</i>	В пікселях визначає розміри горизонтальних відступів (полів) від меж фрейма до об'єктів, що в ньому відображаються
<i>marginheight</i>	В пікселях визначає розміри вертикальних відступів (полів) від меж фрейма до об'єктів, що в ньому відображаються
<i>noresize</i>	Наявність цього параметру забороняє користувачам міняти розміри фреймів.

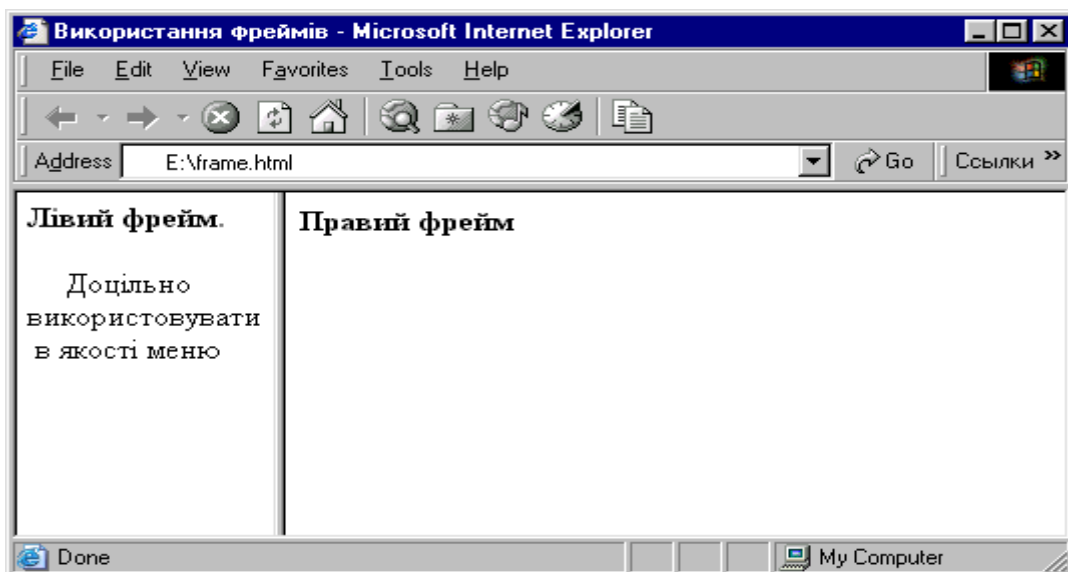


Рис. 1.13 - HTML-документ з фреймовою структурою

2 ЛАБОРАТОРНИЙ ПРАКТИКУМ

2.1 Лабораторна робота № 1.

"Використання ресурсів мережі Інтернет за допомогою браузера Microsoft Internet Explorer"

Мета роботи: опанувати основні прийоми роботи з браузером Microsoft Internet Explorer.

Завдання:

1. За допомогою пошукової системи МЕТА знайти адресу сайту Міністерства освіти та науки України та записати її в папку "Избранное" під назвою МОН України.
2. Визначити, яке кодування використовується на сайті МОН України.
3. Встановити сайт пошукової системи МЕТА в якості домашньої сторінки.
4. Відключити можливість показу рисунків на сайтах.
5. Відключити можливість показу відео-зображень на сайтах.
6. Відключити можливість програвання звуку на сайтах.
7. Відновити програвання відео та звуку.

Методичні вказівки.

1. Відкрити вікно браузера Microsoft Internet Explorer.
2. Використавши команди браузера "Справка → О программе" визначити версію браузера.
3. Налаштувати інтерфейс браузера. Для цього можливо використовувати контекстне меню рядка меню або панелі інструментів браузера (рис. 2.1) та команду "Настройка" контекстного меню. Виконання даної команди призводить до появи вікна настройки панелі інструментів браузера (рис. 2.2).

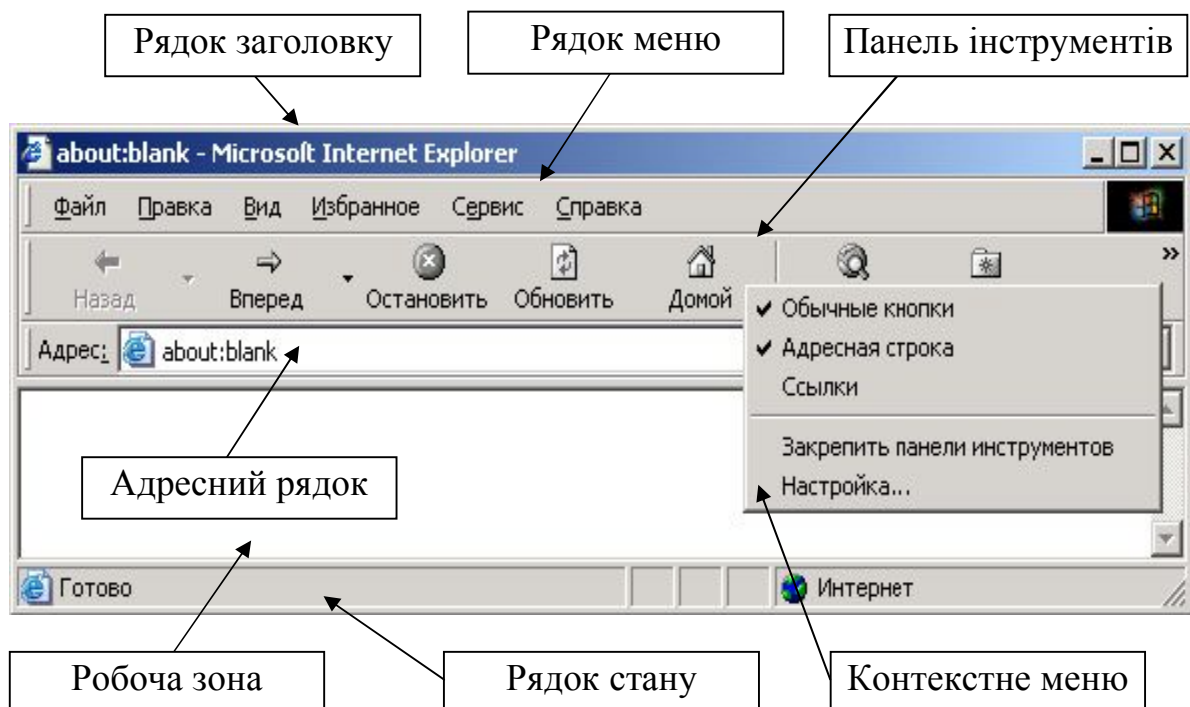


Рис. 2.1. Основні компоненти інтерфейсу браузера Microsoft Internet Explorer

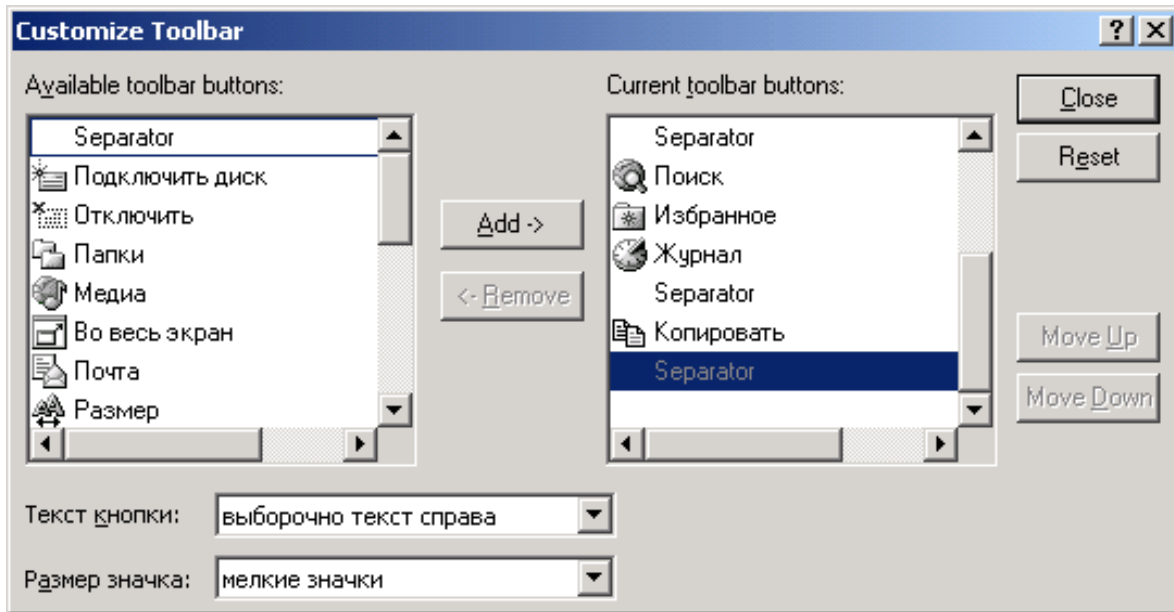


Рис. 2.2. Вікно настройки панели инструментов браузера

4. Використавши команди "Сервис->Свойства обозревателя" відкрити вікно "Свойства обозревателя". Перейти на вкладку "Подключения". За допомогою технічного персоналу встановити з'єднання браузера з мережею Інтернет.

5. Відвідати Web-сайти Кабінету міністрів України (<http://www.kmu.ua>), Верховної Ради України (<http://www.rada.gov.ua>), пошукових систем Meta (<http://www.meta.ua>) та Google (<http://www.google.com.ua>). Для цього слід ввести в адресному рядку відповідну адресу. Відзначимо, що вводити **http://** необов'язково.

6. За допомогою команди "Файл"->"Создать"->"Окно" створити нове вікно браузера та відкрити в ньому сайт конституційного суду України (<http://www.ccu.gov.ua/>).

7. Ознайомитись з можливостями команди "Вид"->"Кодировка". Після цього встановити автоматичний вибір браузером кодування Web-сторінки.

8. Ознайомитись з можливостями команди "Вид"->"Размер шрифта". Встановити найбільш зручний для вас розмір шрифту.

9. Ще раз відвідати згадані сайти за допомогою кнопок "Назад" та "Вперед".

10. Ознайомитись з призначенням кнопок "Обновить", "Домой" та "Стоп".

11. Використавши кнопку "Избранное" записати адреси відвіданих сайтів в теку вибраних адрес.

12. Використавши кнопку "Журнал" здійснити навігацію по сайтам, що вже були відвідані за допомогою даного браузера.

13. Використавши команди "Сервис->Свойства обозревателя" відкрити вікно "Свойства обозревателя". Перейти на вкладку "Общие". Встановити параметри показані на рис. 2.3. Це означає, що браузер буде відкриватись з порожньої сторінки, а адреси переглянутих сторінок будуть зберігатись 20 днів.

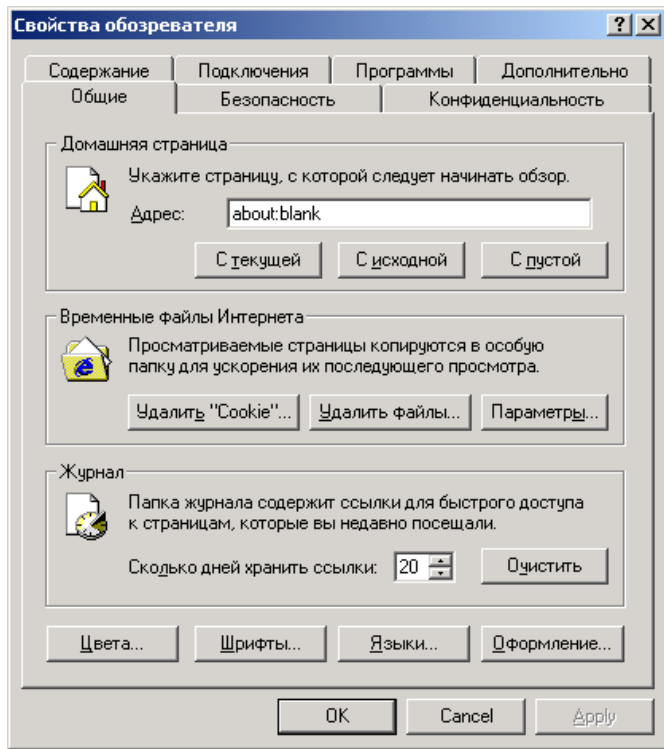


Рис. 2.3. Настройка загалньх параметрів функціонування браузерa

14. Використавши вкладки "Безопасность" та "Конфединциальность" вікна "Свойства обозревателя", ознайомитись з параметрами, що визначають безпечний перегляд різних Web-сайтів.

15. У вікні вікна "Свойства обозревателя" перейти на вкладку "Содержание" та натиснути кнопку "Включить" в полі "Ограничения доступа". За допомогою нового вікна настройки "Ограничения доступа" (рис. 2.4) обмежити доступ до інформації, отриманої з мережі Інтернет.

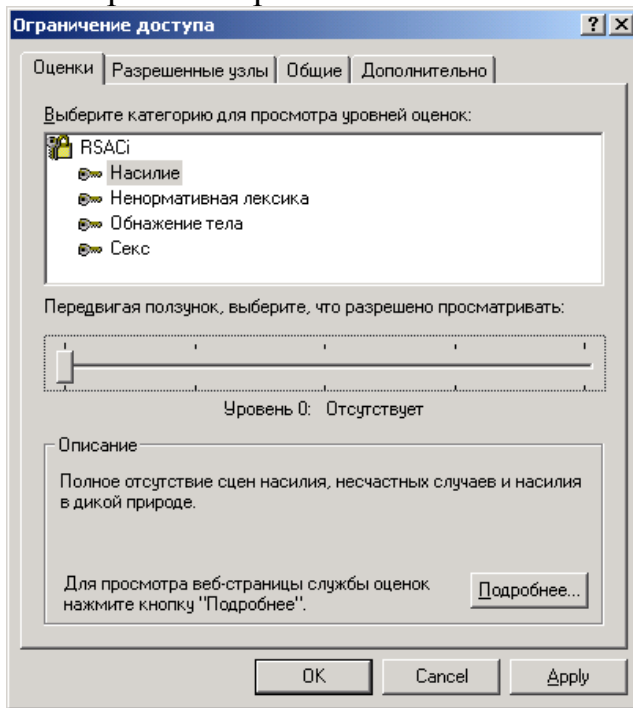


Рис. 2.4. Визначення обмежень доступу до інформації, що відображається в браузері.



16. У вікні вікна "Свойства обозревателя" перейти на вкладку "Дополнительно". Знайти розділ "Мультимедиа". Відмінити відображення рисунків в вікні браузера. Відвідати декілька вже переглянутих сайтів. Впевнитись в тому, що час, потрібний для відображення сайту в вікні браузера, зменшився.

17. За допомогою контекстного меню в місці, призначеному для рисунку, відобразити його на екрані.

18. Відновити відображення рисунків.

19. Освоїти можливості збереження Web-сайтів. Для цього можливо використати команди меню "Файл"-"Сохранить как..." Ознайомитись з різними типами файлів, що можуть бути використані при збереженні інформації із Інтернет.

20. Освоїти можливості збереження фрагментів Web-сайтів.

- Рисунок можливо зберегти за допомогою команди "Сохранить рисунок как..." відповідного контекстного меню.
- Текст зберігаємо за допомогою копіювання виділеного фрагменту в будь-який текстовий редактор.

21. Записати на свій комп'ютер файл, розміщений на FTP-сервері. Для цього:

- Записати в адресному рядку URL визначеного FTP-серверу. Наприклад, ftp://ftp.microsoft.com.
- Для збереження файлу необхідно викликати контекстне меню відповідного елемента у вікні браузера та вибрати команду "Копировать в папку".

Контрольні питання:

1. Яке призначення кнопки "Домой"?
2. Як зберегти тільки текстову частину Web-сторінки?
3. Як повністю зберегти Web-сторінку?
4. Як зберегти рисунок на Web-сторінці?
5. Як зберегти фрагмент тексту на Web-сторінці?
6. Як відмінити можливість відображення рисунків у вікні браузера?
7. Назвіть основні параметри безпеки браузера?
8. Як заборонити доступ до певної інформації, розміщеної в мережі Інтернет?
9. Як змінити кодування Web-сторінки?
10. Як реалізувати доступ до FTP-серверу за допомогою браузера?

2.2. Лабораторна робота №2 "Створення текстової HTML-сторінки"

Мета роботи: освоїти основні прийоми роботи по створенню HTML-сторінки за допомогою текстового редактору "Блокнот" та методику використання тегів форматування тексту HTML-документу.

Завдання:

Оформити у вигляді HTML-документу сторінку тексту, що наведені нижче, відповідного варіанту.

Методичні вказівки.

1. Підготовчі роботи:

- Створити теку, де будуть розміщуватись HTML-файли. Наприклад, на диску Е створити теку HTML.
- Встановити настройки для відображення розширень файлів. Для цього зайти в нашу теку та вибрати команди меню "Сервис" → "Свойства папки". У вікні "Свойства папки" перейти на вкладку "Вид" та зняти перемикач у рядку "Скрывать расширения для зарегистрированных типов файлов" (рис.2.5).

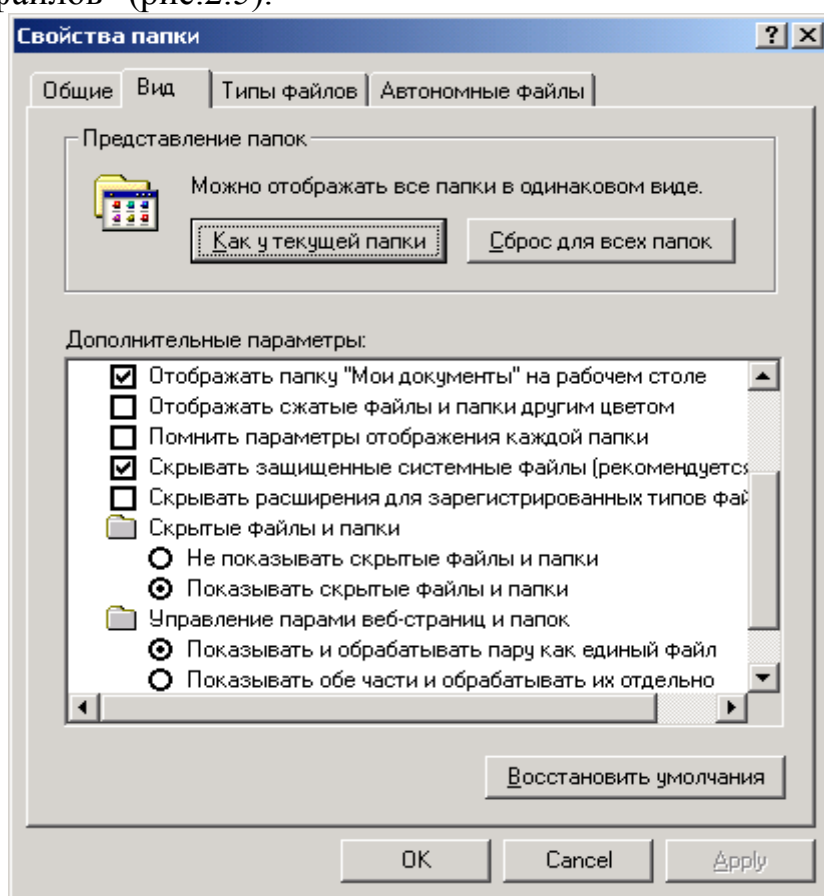


Рис. 2.5. Настройка папки для відображень розширення файлів

2. В теці HTML створити текстовий документ з назвою text.html. При цьому повинно з'явитись вікно запити зміни розширення файлу (рис. 2.6). В цьому

вікні натиснути на кнопку "Да". Іконка, файлу text.html зміниться на таку, що відповідає браузеру.

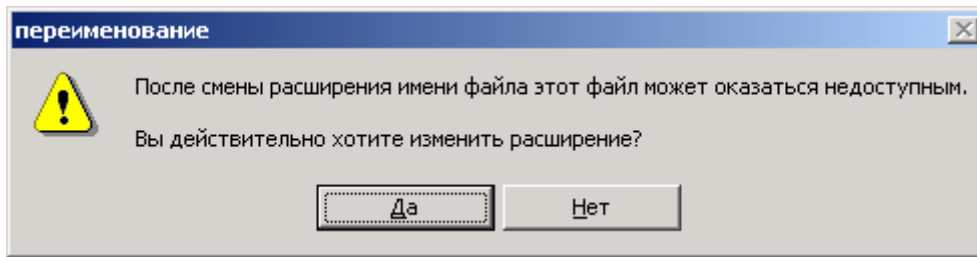


Рис. 2.6. Вікно запиту зміни розширення файлу

3. Відкрити файл text.html за допомогою браузера (рис. 2.7). Відзначити, що на цей момент файл порожній.

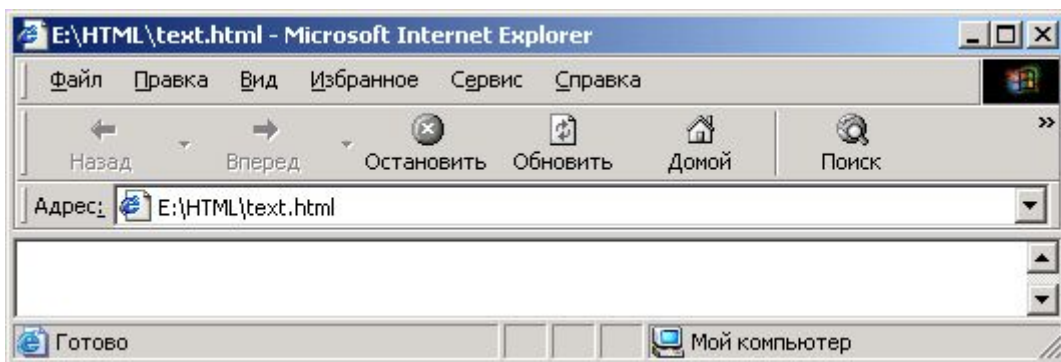


Рис. 2.7. Відображення порожнього файлу text.html у вікно браузера

4. Для запису інформації у файл text.html вибрати наступну команду меню браузера: "Вид" → "Просмотр HTML-кода". У відповідь на це відкривається вікно текстового редактора "Блокнот" з файлом text.html.

5. В редакторі "Блокнот" записати HTML-код, що визначає основні розділи HTML-документу, встановлює кодування Windows-1251 та встановлює назву документу "Лабораторна робота №2". Відзначити, що назва HTML-документу відображається в рядку заголовку браузера.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
      charset=windows-1251">
    <title> Лабораторна робота №2</title>
  </head>
  <body>
  </body>
</html>
```

6. Зберегти в "Блокноті" записаний текст, перейти в браузер та за допомогою кнопки "Обновить" оновити сторінку text.html (рис. 2.8).

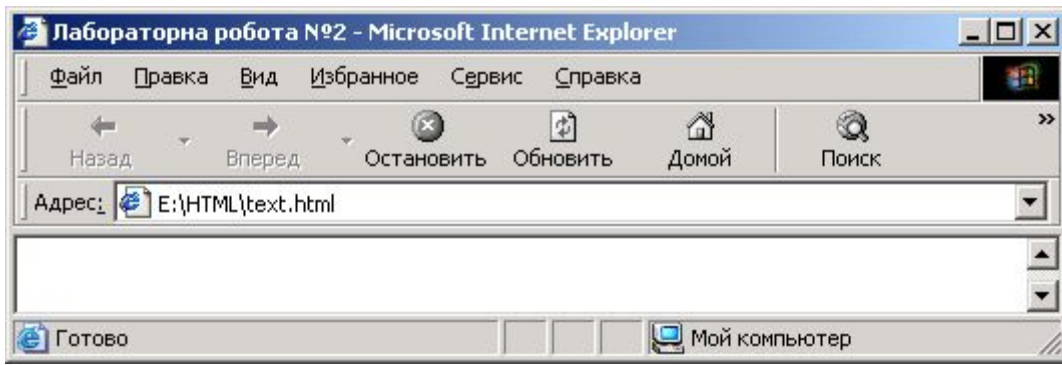


Рис. 2.8. Відображення назви HTML-документу

7. Для визначення тексту напівжирними символами, курсивом та підкресленими символами в тілі HTML-документу записати:

```
<body>
<b>напівжирні символи</b>,
    <i>курсив</i>,
    <u>підкреслені символами</u>,
    <b><i><u>комбінація</u></i></b>
</body>
```

Зберегти документ у "Блокноті" та оновити його в браузері (рис. 2.9.)

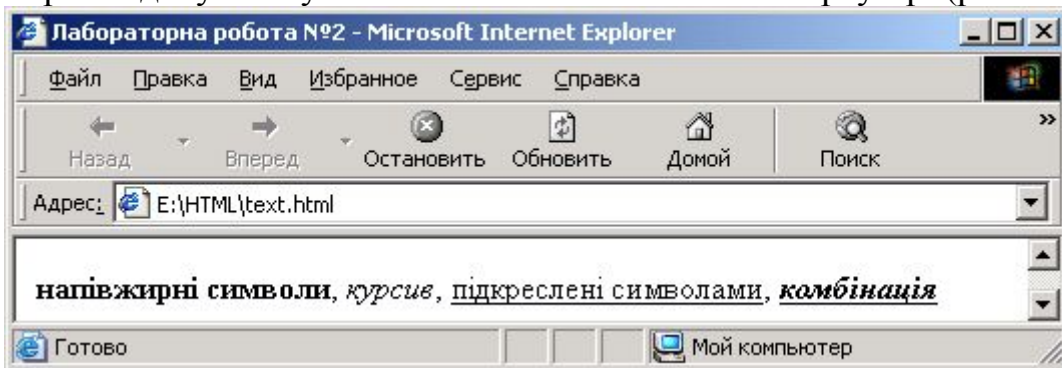


Рис. 2.9. Форматування тексту

8. Додати в HTML-документ формулу $X^2+X^4 = Y_i$. Для цього в кінці тіла документу дописати:

```
...
<br>X<sup>2</sup>+X<sup>4</sup>=Y<sub>i</sub>
</body>
```

Відзначимо, що тег `
` використано для запису формули з нового рядка.

9. Додати в наш документ HTML-код для відображення тексту у вигляді заголовків:

```
...
<h1>Заголовок 1</h1>
<h2 align="center">Заголовок 2. Вирівнювання "по центру"</h2>
<h3 align="right">Заголовок 3. Вирівнювання "по правому краю" </h3>
<h4>Заголовок 4</h4>
<h5>Заголовок 5</h5>
```



Заголовок 6

</body>

Фрагмент вікна браузера з текстом у вигляді заголовків показаний на рис. 2.10.

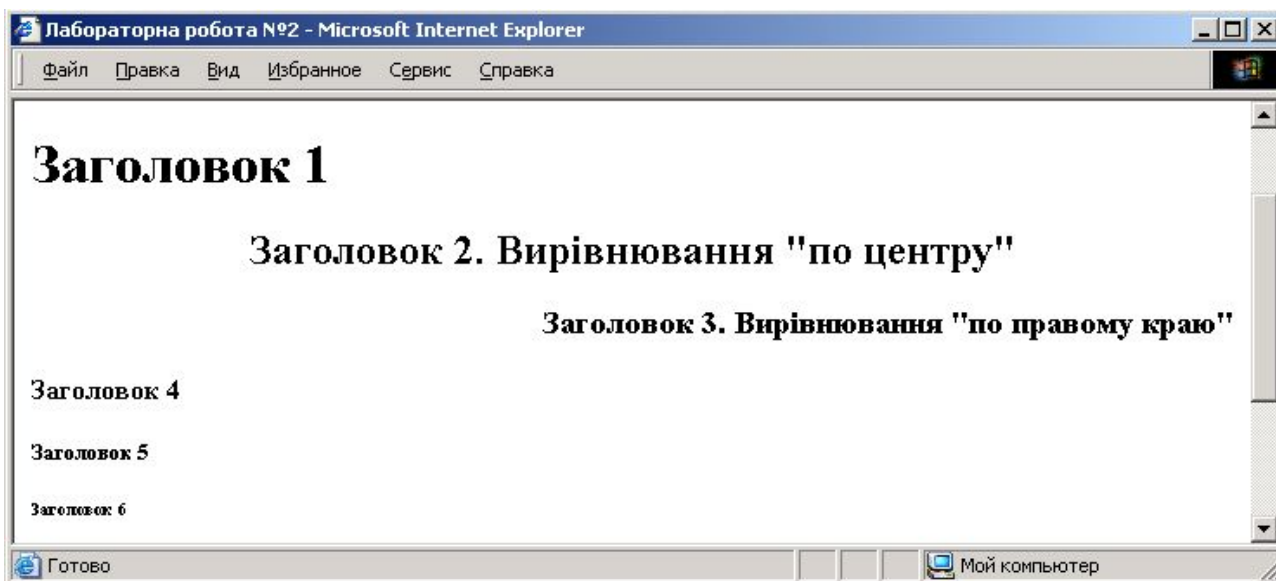


Рис. 2.10. Відображення заголовків

10. Додати в документ три фрагменти тексту червоного кольору. Перший фрагмент записати шрифтом "Arial" розміром 7, другий - шрифтом "GothicG" розміром 3, третій - "Verdana" розміром 1. Для цього записати наступний HTML-код:

```

...
<br><font face="Arial" size="7" color="Red">Перший фрагмент
тексту</font>
<br><font face="GothicG" size="3" color="Red">Другий фрагмент
тексту</font>
<br><font face="Verdana" size="1" color="Red">Третій фрагмент
тексту</font>

```

</body>

Відповідний фрагмент вікна браузера показаний на рис. 2.11.

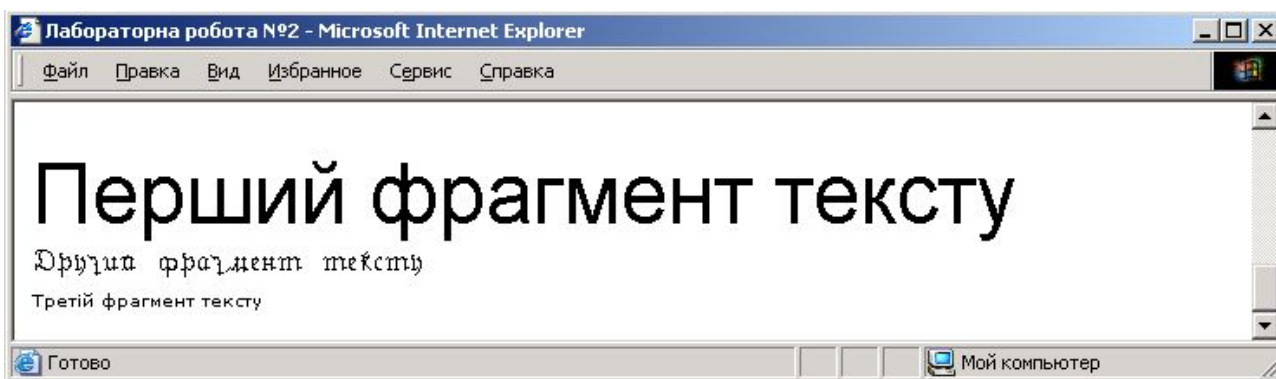
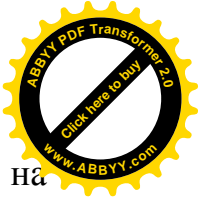


Рис. 2.11. Використання тегу



11. За допомогою тегів абзаців та ліній розділити наш HTML-документ на окремі частини. До того ж визначити, що колір фону сторінки сірий, а колір тексту коричневий. Крім того встановити, що горизонтальні відступи від об'єктів HTML-сторінки до границь вікна браузера дорівнюють 3, а вертикальні 2 пікселі. Для цього змінити HTML-код наступним чином:

```
<body leftmargin=2 topmargin=1 bgcolor="Silver" text="Maroon">
  <p align="right">
    <b>напівжирні символи</b>,      <i>курсив</i>,  <u>підкреслені
    символами</u>,
    <b><i><u>комбінація</u></i></b>
    <br>X<sup>2</sup>+X<sup>4</sup>=Y<sub>i</sub>
  </p>
  <hr width="100%" size="2" noshade>
  <h1>Заголовок 1</h1>
  <h2 align="center">Заголовок 2. Вирівнювання "по центру"</h2>
  <h3 align="right">Заголовок 3. Вирівнювання "по правому краю" </h3>
  <h4>Заголовок 4</h4>
  <h5>Заголовок 5</h5>
  <h6>Заголовок 6</h6>
  <hr width="70%" size="3" noshade>
  <p align="center">
    <br><font face="Arial" size="7" color="Red">Перший фрагмент
    тексту</font>
    <br><font face="GothicG" size="3" color="Red">Другий фрагмент
    тексту</font>
    <br><font face="Verdana" size="1" color="Red">Третій фрагмент
    тексту</font>
  </p>
</body>
```

Відображення даної HTML-сторінки у вікні браузера показано на рис.2.12.

12. Набрати текст свого варіанту і за допомогою відповідних тегів придати йому відповідний вигляд.

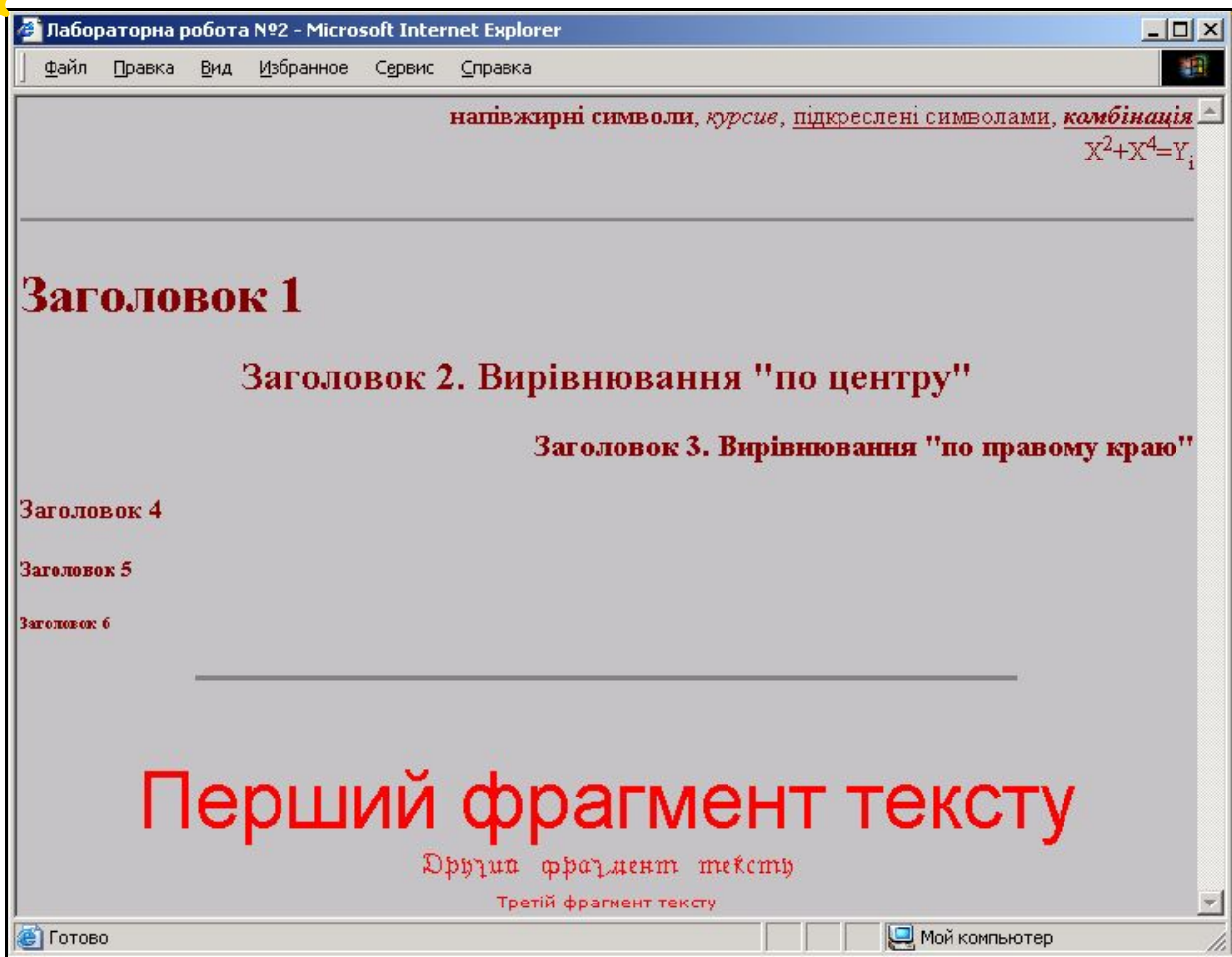
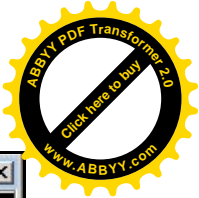


Рис. 2.12. Використання форматування текстової частини HTML-документу

Варіанти завдань до лабораторної роботи № 2

Варіант № 1

*Радість бачити і розуміти
є найпрекрасніший дар природи.
Альберт ЕЙНШТЕЙН*

Загадка небесної блакиті

Чому небо блакитне?...

Немає такої людини, яка не задумалася над цим хоч раз в житті. Пояснити походження кольору неба прагнули вже середньовічні мислителі. Деякі з їх припускали, що синій колір – це істинний колір повітря або якогось-небудь із становлячих його газів.

Інші думали: справжній колір неба чорний – такий, яким воно виглядає вночі. Вдень же чорний колір неба складається з білим – сонячного проміння, і виходить голубий.

Зараз не зустрінеш людину, яка, бажаючи одержати блакитну фарбу, став би змішувати чорну і білу. А був час, коли закони змішення кольорів були ще неясні. Їх встановив всього триста років тому **НЬЮТОН**.



Варіант № 2

Яка історія розвитку парової машини?

Парова машина

Ідея парового двигуна підказала його винахідникам конструкцією поршневого водного насоса, який був відомий ще за часів античності. Вперше модель такої машини була запропонована в 1690 році Папенієм.

Найбільш ширше застосовувалася в першій половині XVIII століття парова машина Ньюкомена, створена в 1711 році. Основний недолік машини Ньюкомена: робочий циліндр в неї був в той же час і конденсатором. **Через це доводилося по черзі то охолоджувати, то нагрівати циліндр, і витрата палива виявлялася дуже великий.**

У 1718 році Бейтон придумав самодіючий розподільний механізм, який автоматично включав або відключав пару і впускав воду. Але принципова схема машини Ньюкомена залишалася незмінна впродовж 50 років, поки її удосконаленням не зайнявся механік університету в Глазго Джеймс Уатт.

Варіант № 3

Скільки буває веселок?

Веселка

Недосвідчений спостерігач бачить звичайно одну веселку, зрідка дві. Причому *друга веселка*, концентрична з *першою*, має кутовий радіус біля 50° і розташовується над *першою*. Друга веселка ширша, блякла, розташування кольорів в неї зворотне *першій* веселці: зовнішня дуга в неї фіолетова, а внутрішня червона.

Найдивовижніше, що більшість людей, що спостерігали веселку багато раз, не бачить, а точніше не помічають додаткових дуг у вигляді ніжних кольорових арок усередині *першої* і зовні *другої* веселок (тобто з боку фіолетових країв веселок).

Ці кольорові дуги (їх звично три-чотири) неправильно названі *додатковими* - насправді вони такі ж основні, як *перша* і *друга* веселки.

Центром веселки є крапка, діаметрально протилежна Сонцю, - антисолярна крапка. Зовнішня дуга веселки червона, за нею йде оранжева, жовта, зелена дуги і т.д., кінчаючи внутрішньою фіолетовою.



Варіант № 4

Вдосконалення системи управління реалізацією продукції (Що таке франчайзинг?)

Франчайзинг – це система договірних відносин між головним підприємством і невеликою діючою або створюваною знов фірмою, що веде роздрібну продажі товарів або послуг.

Головна компанія вимагає ось такого підприємства виплати одноразової первинної винагороди за право оперувати на ринку ось її імені, під її торговою маркою. Така винагорода складає звичайно 10-25 тис. дол., але може бути у декілька разів більше або менше в залежності ось галузі або сфери бізнесу.

В рамках договірної системи головне підприємство здає в оренду підприємству, безпосередньо обслуговуючому споживачів, основні фонди; надає короткостроковий кредит на пільгових умовах; поставляє напівфабрикати або готову продукцію, що підлягає реалізації, і надає інші види послуг. **Дуже багато процвітаючих фірм на Западе вдавалися до франчайзингу і як до засобу маркетингу, і як до елемента фінансової політики.**

Варіант № 5

Управління реалізацією продукції

(Перші нариси)

Основними напрямками вдосконалення системи управління реалізацією продукції є різні нововведення в побудові збутової сети: франчайзинг, сумісний збут, телемаркетинг, а також різні методи просування продукту на ринок: рекламна діяльність, стимулювання збуту, суспільна або публічна популярність (паблісіті) і зв'язок з громадськістю, персональні продажі.

Сумісний збут — це реалізація всієї або значної частини своєї продукції під чужою торговою маркою, використовуючи збутову мережу іншої (звично крупній і відомій компанії) і поступаючись частиною прибутку (до 50%) той компанії, яка надає свою торгову марку.

Переваги методу — не потрібно нести витрати по створенню власної збутової сети (що особливо важливо при роботі на новому ринку), економія на рекламі, підвищення рентабельності порівняно з аутсайдерами на новому ринку (отримання можливості реалізовувати свою продукцію на новому ринку відразу за максимально високою ціною).



Варіант № 6

Криза і крах радянської системи

Реформи Н.Хрущева, відлига (1953-1964 рр.)

Після сталінського режиму час правління Н. Хрущева представлявся яскравим контрастом. Живий, енергійний, захоплюється реформатор генерував ідеї і гасла, що приголомшували систему, розбурхували розум змученого населення. Є зразки словесної гри Н.Хрущева з керівниками західних держав.

Із спогадів посла СРСР в США А.Ф. Добриніна в Карібській кризі:

За довгі 24 роки роботи як посол СРСР в США довелося пережити немало драматичних і напружених подій, якими рясніли радянсько-американські відносини в період "холодної війни".

Мабуть, тим, що запам'ятався була найнебезпечніша Карібська криза 1962 року, що вперше поставив мир на грань ядерної катастрофи.

Варіант № 7

Як виникає веселка?

Звідки береться дивовижне барвисте світло, витікаючий ось дуг веселки?

Всі веселки - це сонячне світло, розкладене на компоненти і переміщене по небозводу таким чином, що он здається витікаючим ось частини небозводу, протилежної той, де знаходиться Сонце.

Наукове пояснення веселки вперше дав Рене Декарт в 1637 р. Декарт пояснив веселку на підставі законів заломлення і віддзеркалення сонячного світла в краплях випадного дощу. У той час ще не була відкрита дисперсія — розкладання білого світла в спектр при заломленні. Тому веселка Декарта була білою.

Через 30 років **Ісаак Ньютон**, що відкрив дисперсію білого світла при заломленні, доповнив теорію Декарта, пояснивши, як заломлюються кольорові лучі в краплях дощу. По образному виразу американського ученого А. Фразера: "**Декарт** повісив веселку в потрібному місці на небозводі, а **Ньютон** розцвітив її всіма фарбами спектру".



Варіант № 8

Розмір і форма крапель і їх вплив на вигляд веселки

Розрахунки по формулах дифракційної теорії, виконані для крапель різного розміру, показали, що весь вид веселки: ширина дуг, наявність, розташування і яскравість окремих колірних тонів, положення додаткових дуг дуже сильно залежать ось розміру крапля дощивши.

Радіус крапля 0,5-1 рр.

Зовнішній біля основної веселки яскравий, темно-червоний, за ним йде світло-червоний і далі чергуються всі барви веселки. Особливо яскравими здаються фіолетовий і зелений. Додаткових дуг багато (до п'яти), в них чергуються фіолетово-рожеві тони із зеленими. Додаткові дуги безпосередньо примикають до основних веселок.

Радіус крапля 0,10-0,15 рр.

Червоного кольору в основній веселці більше немає. Зовнішній біля веселки оранжевий. У іншому веселка добре розвинена. Додаткові дуги стають все більш жовтими. Між ними і між основною веселкою і першої додаткової з'явилися просвіти.

Варіант № 9

ВІДЛУННЯ ЧОРНОБИЛЯ

26 квітня 1986 року відбувся вибух на 4-у блоці Чорнобильської АЕС.

Чорнобиль на століття залишиться в пам'яті людства символом: страху, ганьби і скорботи. На **Западе** (за винятком Франції) відмовилися ось програм ядерного будівництва через різко збільшені витрати, дешевші технології і опозицію, що посилюється, в суспільстві, а **Схід** став плацдармом для нових недоцільних інвестицій в нові заводи і модернізацію старих.

Наслідки катастрофи - медичні, економічні, екологічні і психологічні - набагато більш серйозні по своїх просторових і тимчасових масштабах, чим це признається офіційно.

Російська, білоруська і українська статистика стану здоров'я населення забруднених територій показує значне збільшення числа багатьох захворювань. **МАГАТЕ** продовжує цинічно зменшувати масштаби трагедії. Чи є надія на поліпшення?



Варіант № 10

Аварія на Чорнобильській АЕС і її наслідки

*Дуже часто у наш час героїзм виявляється вимушеним.
Хоча пройшло вже досить багато часу, але урок
Чорнобиля - економічний, етичний, політичний -
повинен залишитися в кожному з нас.*

Хроніка подій одної ночі:

1 час 23 хвилини 03 секунди: Чергування по пожежній охороні АЕС ніс третій караул. Цілий день караул проводив час на п'ятому енергоблоці, що будується, відповідно до звичного розпорядку: теоретичні заняття в учбовому класі, практичні - під керівництвом лейтенанта **Володимира Правіка**.

1 час 30 хвилин: До станції під'їхала машина **Правіка**. По зовнішніх ознаках лейтенант визначив вогнище горіння і організував гасіння пожежі з боку машинного зал і з боку реакторного відділення - для гасіння вогню на кривлі. Як пізніше з'ясувалося, це було єдине вірне рішення. Завдяки ньому і вдалося запобігти розповсюдженню пожежі.

Варіант № 11

"Болить душа по Чорнобилью..."

Наслідки Чорнобильської трагедії

Чорнобиль - невелике, миле, провінційне українське містечко, потопаюче в зелені, все у вишнях і яблунях. Польотом тут любили відпочивати багато: кияни, москвичі, лєнінградці. Приїжджали сюди ґрунтовно, часто на всі літа, готували на зиму варива, збирали гриби, загоряли на сліпуче чистих піщаних берегах Київського моря, ловили рибу.

26 квітня 1986 року відбулася подія, що сколихнула всю країну і весь світ, торкнулася долі мільйонів людей, - на Чорнобильській АЕС відбулася найбільша за всю історію атомної енергетики аварія.

В результаті теплового вибуху реактора четвертого блоку **Чорнобильської АЕС** були пошкоджені і розгерметизовані тепловиділяючі елементи, що містять ядерне паливо і радіоактивні продукти розподілу, що накопичилися за час роботи реактора. Відбувся короткочасний викид **радіоактивних продуктів** в атмосферу із зруйнованого реактора, а потім протягом багатьох днів в атмосферу виділялися газоподібні і аерозольні продукти, утворюючи радіоактивний струмінь.



Варіант № 12

Міражі

Що таке міраж?

Міражі – оптичний ефект, що обумовлений заломленням світла при проходженні через шари повітря різної густини і виражається у виникненні уявного зображення.

Видалені об'єкти при цьому можуть виявитися піднятими або опущеними щодо їх дійсного положення; можуть бути спотворені і придбати неправильні, фантастичні форми. **Міражі** часто спостерігаються в умовах жаркого клімату, наприклад, над піщаними рівнинами.

Звичні **нижні міражі**, коли віддалена, майже рівна поверхня пустелі набуває вигляд відкритої води, особливо якщо дивитися з невеликого піднесення або просто знаходиться вище за шар нагрітого повітря. Подібна ілюзія звичайно виникає на нагрітій асфальтованій дорозі, яка далеко попереду виглядає як водна поверхня. **Насправді ця поверхня є віддзеркаленням неба.**

Варіант № 13

Оптичні явища в атмосфері

Деякі оптичні явища в атмосфері (наприклад, свічення і найпоширеніше метеорологічне явище – блискавка) мають електричну природу.

Вогні святого Ельма.

Вогні святого Ельма – блідо-голубі або фіолетові кисті, що світяться, завдовжки ось 30 см до 1 м і більш, звично на верхівках щогл або кінцях рей судів, що знаходяться в морі.

Іноді здається, що весь такелаж судна покритий фосфором і світиться. **Вогні святого Ельма** деколи виникають на гірських вершинах, а також на шпильях і гострих кутах високих зданій. Вони є електричними розрядами на кінцях електропровідників, коли в атмосфері навколо них сильно підвищується напруженість електричного поля.

Блукаючі вогники

Це слабке свічення голубуватого або зеленуватого кольору, яке іноді спостерігається на болотах, кладовищах і в склепах; виглядає як підведене приблизно на 30 см над землею полум'я свічки, що спокійно горить, на мить зависаючу над об'єктом.



Варіант № 14

Загальна характеристика атмосфери Землі

Що таке атмосфера?

АТМОСФЕРА - газова оболонка, що оточує небесне тіло. Її характеристики залежать ось: розміру, маси, температури, швидкості обертання і хімічного складу даного небесного тіла, визначаються історією його формування починаючи з моменту зародження.

Атмосфера Землі утворена сумішшю газів, званою повітрям. Її основні складові – азот і кисень в співвідношенні приблизно 4:1. Самий нижній, шар атмосфери особливо важливий для людини, яка мешкає в місці контакту твердої, рідкої і газоподібної оболонок Землі.

На людину надає дію головним чином стан нижні 15-25 км атмосфери, оскільки саме в цьому нижньому шарі зосереджена основна маса повітря. Стан верхніх шарів атмосфери, розташованих на висотах ось 60 до 300 і навіть 1000 км ось поверхні Землі, також змінюється. Тут розвиваються сильні вітри, шторми і виявляються такі дивовижні електричні явища, як полярні сяйва.

Варіант № 15

АНОМАЛЬНЫЕ ЗОНЫ

Ігор Царев. Планета примар

Мова піде про аномальні зони, зони "звичних" і зони "дивні" (так, серед аномальностей попадаються свого роду аномальності в квадраті). Але спочатку - про "звичайні" аномальні зони.

"Космонавти, наприклад, спостерігаючи Землю з орбіти, помітили в океанах гігантські водяні горби і западини з перепадом висот в декілька десятків метрів. Вони стійкі в часі, а значить, це не хвилі. І, до речі, один з таких таємничих "провалів" знаходиться в районі сумно відомого Бермудського трикутника.

Багато невтішних прізвиськ придумали моряки цій ділянці Атлантичного океану: "Диявольське море", "Кладовище Атлантики", "Море проклятих", "Адов круг"...

З легкої руки американця **Вінченца Гаддіса**, що написав в 1964 році статтю "Смертоносний Бермудський трикутник", саме така назва феномена стала загальною визнаною.



Варіант № 16

Вільям Шекспір

(1564-1616)

*Нам невідома точна дата народження **Вільяма Шекспіра**, проте метричні церковні записи свідчать у тому, що он був хрещений в соборі **Святої Трійці в Стретфорде-на-Двоне, Англія, 26 квітня 1564 року.***

Його батько, **Джон Шекспір**, досяг успіху в торгівлі і в 1568 році став мером **Стретфорда**. Його мати, **Марія**, була дочкою процвітаючого поміщика. Історії мало відомо в шкільних роках **Шекспіра**: відомо лише, що он знав небагато по-латині і ще менше по-грецьки.

У університеті он не вчився. Згідно двом поширеним легендам молодий **Шекспір** був спочатку помічником м'ясника, а згодом йому довелося покинути **Стретфорд**. Достовірно відомо, що 27 листопаду 1582 року 18-річний **Шекспір** одружувався.

Варіант № 17

Вільям Шекспір (1564-1616)

(Англійська література)

***Шекспір** народився 23 квітня 1564 року в сім'ї заможного городянина, ремісника і торговця **Джона Шекспіра** в **Стратфорде**, на березі ріки **Ейвон**. У юності **Уільям** вчився в граматичній школі, де вивчав латинську і грецьку мови, історію і літературу.*

Містечко, де жив **Шекспір**, було хоча і маленьким, але стояло на перетині торгових шляхів; городяни велили активний спосіб життя. У 1553 році був вибраний місцевий орган самоврядування - корпорація городян, яка вибрала свою *Раду*, що відала всіма справами міста, навіть модою і стилем одягу. Якийсь час главою *Ради* був батько **Шекспіра**.

Закінчивши школу, **Вільям** допомагав батькові, якийсь час працював вчителем, а у вісімнадцятирічному віці одружувався на дочці багатого фермера **Ганні Хетевей**, яку знав з самого дитинства.



Варіант № 18

Ресурси

Джерела дорогого виробництва – це ресурси, яких має в своєму розпорядженні суспільство.

Ресурси виробництва – це сукупність тих природних, соціальних і духовних сил, які можуть бути використані в процесі створення товарів, послуг і інших цінностей.

Ресурси прийнято ділити на чотири групи: **природні** – потенційно придатні для застосування у виробництві природні сили і речовини, серед яких розрізняють "невичерпні" і "вичерпані"; **матеріальні** – всі створені людиною засоби виробництва, які самі є результатами виробництва; **трудові** – населення в працездатному віці; **фінансові** – грошові кошти, які суспільство в змозі виділити на організацію виробництва.

Значущість окремих видів змінювалося по мер перепоходка ось доіндустріальної до індустріальної, і ось її – до постіндустріальної технології. У доіндустріальному суспільстві пріоритет належав природним і трудовим ресурсам, в індустріальному – матеріальним, в постіндустріальному – інтелектуальним і інформаційним ресурсам.

Варіант № 19

МАТЕРІАЛЬНІ РЕСУРСИ І ЇХ РАЦІОНАЛЬНЕ ВИКОРИСТОВУВАННЯ

Показники використання матеріальних ресурсів

"Економіка" зростає з "виробництва", яке утворює матеріальну основу економіки. Розглянемо докладніше матеріальні ресурси.

Матеріальні ресурси – это сировина, матеріали, паливо, енергія. Вдосконалення нормування, економія і раціональне використання матеріальних ресурсів було і є найважливішою народногосподарською проблемою на всіх етапах розвитку нашої економіки.

Але все таки нашу економіку можна назвати марнотратною: ми витрачаємо на одиницю промислової продукції в 1,5-2 рази більше різних матеріальних ресурсів в порівнянні з передовими капіталістичними країнами. **Причина цього – недосконалість діючої економічної системи, організаційно-економічних умов виробництва, або розподілу і обігу засобів виробництва?**



Варіант № 20

Шкода куріння

Куріння - одна з щонайшкідливіших звичок.

*Дослідженнями доведений, в чому **шкода куріння**. У димі тютюну міститься більше 30 отруйних речовин: нікотин, вуглекислий газ, окисел вуглецю, синильна кислота, аміак, смолянисті речовини, органічні кислоти і інші.*

1-2 пачки сигарет містять смертельну дозу нікотину. **Курця** рятує, що ця доза вводиться в організм не відразу, а дробово. Статистичні дані говорять: в порівнянні з некурящими **длительнокурящие** в 13 разів частіше захворюють стенокардією, в 12 разів - інфарктом міокарду, в 10 разів - язвою шлунку. *Курці складають 96 - 100% всіх хворих на рак легенів. Кожен сьомий той, що довгий час палить хворіє на облітеруючий ендартерійтом - важким недугою кровоносних судин.*

***Тютюнові вироби** готуються з висушеного листа тютюну, яке містить білки, вуглеводи, мінеральні солі, клітковину, ферменти, жирні кислоти і ін. Серед їх важливо відзначити дві групи речовин, небезпечні для людини, - **нікотин і ізопреноїди**.*

Варіант № 21

Фалес Мілетський

(ок. 624 - ок. 546 до н.е.)

Фалес Мілетський - грецький філософ і математик з **Мілета**. Представник іонічної натурфілософії і засновник мілетської школи. Вважався одним з семи мудреців **Греції**. У **Єгипті** займався вивченням причин наповнення, знайшов спосіб вимірювання висоти пірамід.

За словами Геродота, **Фалес** передбачив сонячне затемнення, що спостерігалось 28 травня 585 до н.е. Рахував матерію одушевленою. **Намагаючись визначити основу матеріального миру, дійшов висновку у тому, що нею є вода.**

Фалес - старогрецький філософ, засновник **Мілетської школи філософії**. Хоча прийнято вважати, що західна філософія починається з греків проте перші філософські системи виникли не в самої **Греції** а на західному побережжі **Малої Азії** - в іонійських городах, які були засновані греками і в яких раніше, ніж в самої Греції одержали розвиток промисловість, торгівля і духовна культура. Вперше філософські переконання виникли в **Мілеті** в VI-V століттях до Р.Х. Мілет у той час був найбільшим зі всіх малоазійських грецьких міст.



Варіант № 22

Великі математики

Нільс Абель

Одним з найвидатніших математиків XIX століття був норвезький учений Нільс Абель. Народився **Абель** в 1802 році. Тринадцяти років он був відданий в училищі. **Видатні здібності до занять математикою в Абеля виявилися, коли йому було 16 років.** У цьому віці он полюбив математику і почав наполегливо займатися нею. Після закінчення училища: у 1821г. **Абель** поступив в університет і зразу ж звернув на себе увагу як талановитого математика.

У університетські роки он зробив своє видатне відкриття про рівняння алгебри вищих ступенів. Після закінчення університету Абель зробив поїздку у Берлін і Париж.

За своє коротке життя Абель вніс такий внесок у розвиток математики, який дає право вважати його одним з найбільших математиків. Але відкриття Абеля не були зрозумілими і не оцінені його сучасниками.

Варіант № 23

Японія

Японія - стародавня держава.

Японія - один на економічних лідерів сучасного миру. По виробництву цілого ряду промислових товарів (автомобілів, судів, відеомагнітофонів, промислових роботів і ін.) ця країна займає провідні позиції в світі.

Розташована на чотирьох крупних островах : **Хоккайдо, Хонсю, Сикоку** і **Кюсю** і численних дрібних, поблизу східного побережжя *Євразії*, країна омивається із заходу **Японським морем**, зі сходу - **Тихим океаном**. Острівне положення позначається на розвитку морського господарства, характері внутрішніх і зовнішніх економічних зв'язків і т.д. Столиця **Японії** — **Токіо** знаходиться на одній широті з південним краєм **Туркменії**.

*Географічне положення **Японських островів** на схід ось материка визначило і образну назву країни — Країна висхідного сонця.*



Варіант № 24

ГЕНІАЛЬНІ МАТЕМАТИКИ

Еваріст Галуа (1811 – 1832)

Сумним і коротким було життя геніального французького математика – **Еваріста Галуа (1811 – 1832)**. Так і не дістав он спеціальної математичної освіти, але виняткові математичні здібності дозволили йому зробити чудові відкриття. Математичне дарування **Галуа** виявилось дуже рано. **Основні результати своєї чудової теорії, названої його ім'ям, геть одержав вже у віці 16-18 років.**

*За своє дуже коротке життя (21 ч) **Галуа** заклав основи сучасної алгебри. Створена їм теорія рівнянь алгебри вищих ступенів зробила сильний вплив не тільки на розвиток алгебри, але і всієї математики.*

Математичні відкриття Галуа при його житті не були визнані. Свої роботи **Галуа** двічі представляв в **Паризьку академію наук**, але навіть такі крупні математики того часу, як **Коші, Фурье і Пуассон**, не могли понять значення його відкриттів. Роботи **Галуа** в Паризькій академії вдавалися до забуття.

Варіант № 25

ПРИРОДНІ РЕСУРСИ ЛУГАНСЬКОЇ ОБЛАСТІ

Рельєф

Поверхня області є хвилястою рівниною, що підноситься ось долини **Северського Донца** на північ і на південь, де розташований **Донецький кряж**, утворений могутніми товщами осадкових порід стародавніх морів, що колись існували тут. Вища крапка: **Могила Мечеть** - 367 р.

*Найхарактернішою межею кряжа є чергування горбистих вододільних. У донесе річок **Міуса і Нагольной** висота **Донецького кряжа** знижується; південні схили переходять в **Приазовську берегову рівнину.***

На північ зниження відбувається поступово і до Северському Донцу обривається крутим уступом, утворюючи живописний правий берег річки. У лівобережній частині області тягнеться **Старобельська** рівнина. На крайній півночі в межі області заходять відроги **Среднерусської** піднесеності.



Варіант № 26

ЛУГАНСЬКА ОБЛАСТЬ

Рослинність

Рослинність в результаті діяльності людини піддалася великим змінам. Велика частина території області розорана, лише по схилах ярів, в річкових долинах і в заповідниках (*Стрельцовська степ, Провальська степ*) збереглися ділянки степової рослинності; тут виростає більше тисячі видів різних рослин.

Переважають ліси байрачного типу. Вони відрізняються істотною різноманітністю: у них налічується близько 50 порід дерев і чагарників. Переважають такі породи - дуб, береза, ясен.

Тваринний світ

Територія *Луганської області* входить в східну степову зону *України*. Фауна області представлена головним чином степовими і деякими лісовими тваринами, проте, дика фауна бідна. З хижаків зустрічаються: **вовк, лисиця, снітовидний собака, ласка** і ін.

Варіант № 27

Південно-східна частина України

Донецька область

Донецька область розташована в степовій зоні південно-східної частини *України*. На південному заході і западе вона граничить з *Дніпропетровською і Запорізькою*, на північному заході — з *Харківською*, на північному сході — з *Луганської* областями *України*, на сході — з *Ростовською* областю *Російської Федерації*, а з півдня омивається **Азовським морем**.

Донецька область має унікальні природні багатства, вигідне географічне положення, могутній промисловий потенціал, сучасний рівень науки.

Область має: добре розвинену соціальну інфраструктуру, висококваліфіковані трудові ресурси. На її долю доводиться більше половини здобутого в *Україні* вугілля, виробленого *готового прокату, коксу, чавуну, стали* і практично весь об'єм випуску *прокатного оснащення*. У *Донецькій області* зосереджені найважливіші для господарського комплексу *України* підприємства кольорової металургії по виробництву свинцю, цинку, ртуті, рідкоземельних металів.



Варіант № 28

РЕГІОНИ УКРАЇНИ

Донбас

Величезне значення в економіці *України* має агропромисловий комплекс. Донбас займає перше місце в державі по валовій продукції сільського господарства. *Економічний потенціал Донетчини надає можливість для розвитку ділового різнопланового партнерства.*

Донецька область має різноманітні види транспорту, в першу чергу, найбільшу в *Україні* Донецьку залізну дорогу, на долю якої випадає до 40% загальнодержавних перевезень. Існує густа мережа автомобільних доріг загальнодержавного і місцевого значення. У *Донецьку* розташований міжнародний аеропорт

Екологічна політика, яка проводиться в *Донецькій області*, направлена на збереження і відновлення навколишнього середовища регіону, здійснення принципів раціонального природокористування.

Варіант № 29

ХІМІЯ НАВКОЛО НАС

Юрій ЗОЗУЛИН

Всюди, куди б ні обернув свій погляд, нас оточують предмети і вироби, виготовлені з *речовин і матеріалів*, які одержані на хімічних заводах і фабриках. *Крім того, в повсякденному житті, сам того не підозрюючи, кожна людина здійснює хімічні реакції.*

Наприклад, умивання з **милом**, прання з використанням **миючих засобів** і ін. При опусканні **шматочка лимона** в **стакан гарячого чаю** відбувається **ослаблення забарвлення** – **чай** тут виступає в ролі **кислотного індикатора**, подібного **лакмусу**

Запалюючи **сірник**, замішуючи **пісок і цемент** з **водою** або **гасячи водою ванно**, обпалюючи **цеглину**, ми здійснюємо справжні, а іноді і досить **складні хімічні реакції**. Пояснення цих і других широко поширених в житті людини хімічних процесів – **доля фахівців**.



Варіант № 30

Хімія в сучасному світі

Савінкіна Олена Володимирівна

Хімія тісно пов'язана з **екологією** – наукою про відносини рослинних і тваринних організмів і їх співтовариств між собою і навколишнім середовищем, оскільки і самі живі організми, і оточуюча їх середовища складаються з хімічних речовин.

*На ранніх етапах історії людства діяльність людей майже не зачіпала глибини Землі. Але з початком бурхливого розвитку **промисловості** людині стало бути потрібним все більше і більше **хімічних речовин**, що здобуваються із земних надр.*

Відходи підприємств і транспорту істотно порушують природні атмосферні процеси в індустріальних районах. Слідством викидів оксидів сірі і азоту є утворення кислотних дощів, які випадають на поверхню Землі не тільки в околицях промислових підприємств, але і в сусідніх з ними **районах**. У великих містах і промислових центрах звичним явищем стало утворення смогу, до складу якого входять **дим**, **туман**, **пил** і різні шкідливі для здоров'я **газоподібні речовини**.

Контрольні питання:

1. Як визначити розмір вертикального відступу від тексту до границі браузера?
2. Як визначити розмір горизонтального відступу від тексту до границі браузера?
3. Як визначити колір тексту для всього документу?
4. Як визначити колір фрагменту тексту?
5. Як визначити шрифт, яким текст буде відображатись у вікні браузера?
6. Як записати тег лінії з товщиною в 5 пікселів?
7. Чи можливо задати лінію товщиною в 200 пікселів?
8. Як записується тег заголовку 5 розміру з вирівнюванням по центру?
9. Як записується тег абзацу з вирівнюванням по лівому краю?
10. Яким чином задати обов'язкове відображення фрагменту тексту з нового рядка?



2.3. Лабораторна робота №3 "Використання рисунків, відео та звуку на HTML-сторінці"

Мета роботи: освоїти методику використання рисунків на HTML-сторінці.

Завдання:

Оформити у вигляді HTML - документу фотоальбом з теми тики, вказаної викладачем.

Методичні вказівки.

1. Копіювати в теку HTML видані викладачем графічні файли 1.jpg та 1a.gif.
2. В теці HTML створити текстовий документ з назвою picture.html, відкрити його за допомогою браузера та перейти до редагування HTML-коду.
3. Створити заготовку для HTML-коду Web-сторінки:

```
<html>  
  <head>  
    <title>Рисунки</title>  
    <META http-equiv=Content-Type content="text/html;  
      charset=windows-1251">  
  </head>  
  <body>  
  </body>  
</html>
```

4. Додати HTML-код для відображення на сторінці чотирьох однакових рисунків з різними розмірами:

```
<body>  
  <h2 align=center>Зміна розмірів рисунків</h2>  
    
    
    
    
</body>  
</html>
```

5. Перегляд сторінки в браузері повинен відповідати рис. 2.13.

6. Розглянути можливість використання границь навколо рисунків. Для цього додамо наступний HTML-код:

```
...  
<h2 align=center>Границі рисунків</h2>  
  
  
  
</body>
```

Відповідне відображення у вікні браузера показане на рис. 2.14.



Рис. 2.13. Рисунки з різними розмірами

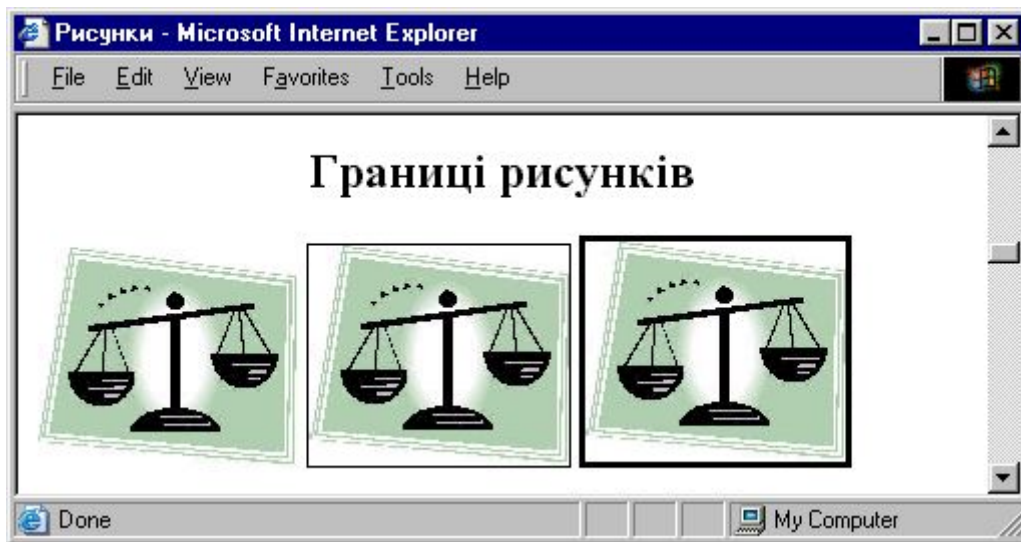


Рис. 2.14 Використання границь рисунків

7. Розглянути можливість визначення альтернативного тексту в тегах рисунків. Для цього модифікувати останній фрагмент HTML-коду:

```
...  

```

```


</body>
```

Відключити показ рисунків у вікні браузеру та провести оновлення нашої сторінки (рис. 2.15).

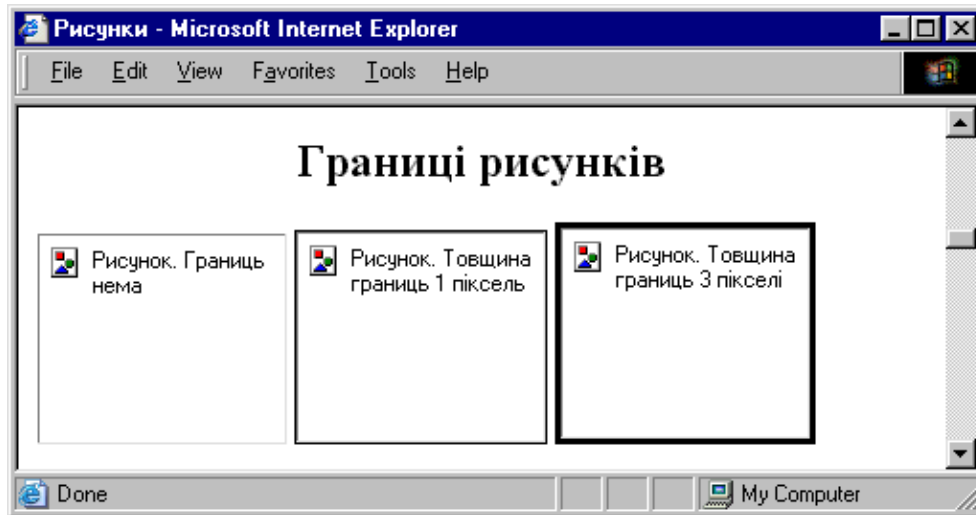


Рис. 2.15. Відображення альтернативного тексту

8. Відновити показ рисунків в браузері та провести оновлення нашої HTML-сторінки.

9. Розглянути можливості вертикального вирівнювання рисунків відносно тексту.

– Додати HTML-код для визначення тексту та рисунків, вирівняних відносно верхньої межі:

```
...
<h2 align=center>Вирівнювання рисунків відносно тексту </h2>
<h3 align=center>Вирівнювання рисунків відносно верхнього межі рядка
</h3>
<p>


По верхній межі найвищого елемента рядка
</p>
<p>


По верхній межі найвищого текстового елемента рядка
</p>
</body>
```

Відповідне вікно браузера показано на рис. 2.16.

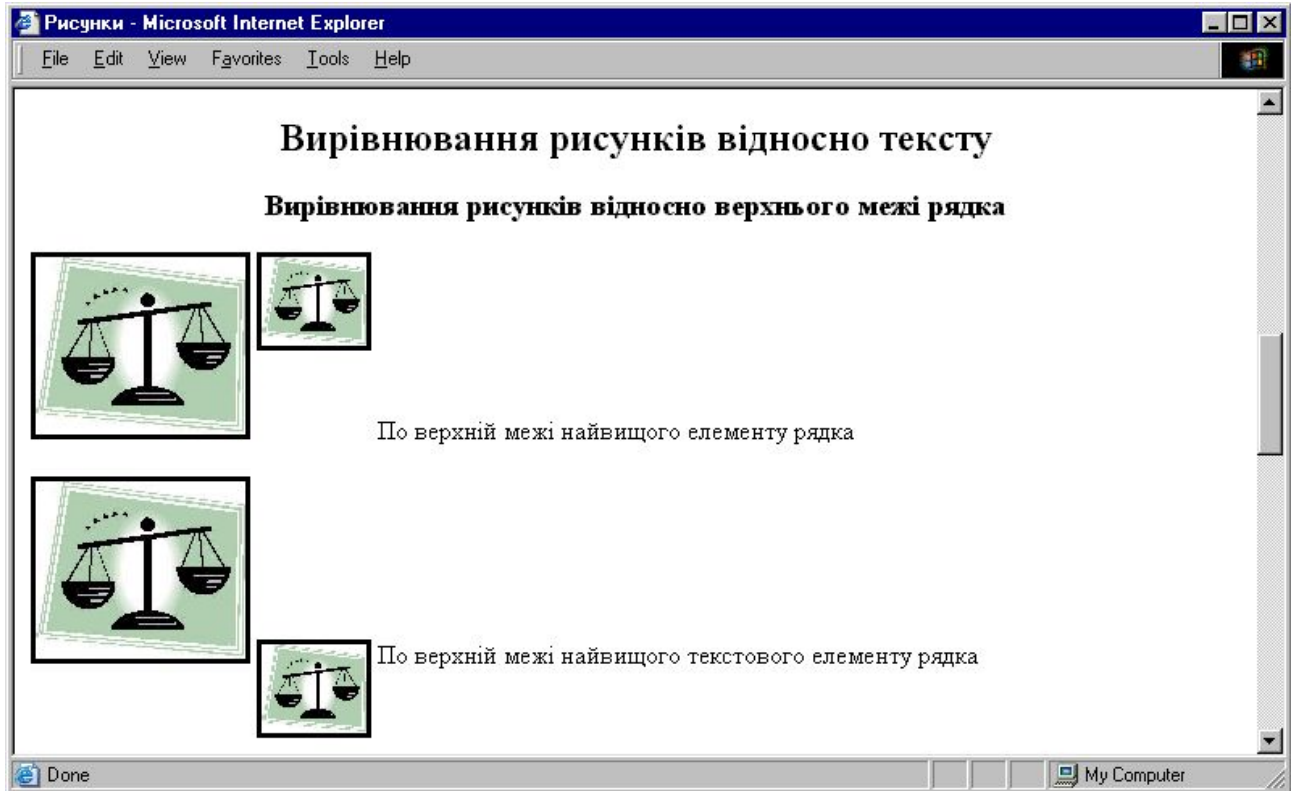


Рис. 2.16. Вирівнювання рисунків відносно верхньої межі

– Додати HTML-код для визначення тексту та рисунків, вирівняних відносно середини рядка:

...

```
<h3> Вирівнювання рисунків відносно середини рядка </h3>
```

```
<p>
```

```

```

```

```

По базовій лінії рядка

```
</p>
```

```
<p>
```

```

```

```

```

По середині рядка

```
</p>
```

```
</body>
```

Відповідне вікно браузера показано на рис. 2.17.



Рис. 2.17. Вирівнювання рисунків відносно середини рядка

– Додати HTML-код для визначення тексту та рисунків, вирівняних відносно нижньої межі:

```

...
<h3 align=center> Вирівнювання рисунків відносно нижнього краю рядка
</h3>
<p>


Вирівнювання рисунків по базовій лінії рядка
</p>
<p>


Вирівнювання рисунків по нижній межі рядка
</p>
</body>

```

Відповідне вікно браузера показано на рис. 2.18.



Рис. 2.18. Вирівнювання рисунків відносно нижньої межі рядка

10. Розглянути можливість визначення "плаваючих" рисунків, тобто рисунків, вирівняних по лівому або правому краю рядка. Для цього додати HTML-код:

```

...
<h3 align=center>"Плаваючі" рисунки</h3>
<p>

По лівому краю рядка
</p>
<br><br><br><br><br>
<p>

По правому краю рядка
</p>
</body>

```

Відповідне вікно браузера показано на рис. 2.19.



Рис. 2.19. "Плаваючі" рисунки

11. Розглянути можливості відокремлення тексту від рисунків за рахунок визначення горизонтальних та вертикальних відступів. Для цього додати HTML-код:

```

...
<h2 align="center">Відокремлення рисунків від тексту</h2>

Відстань до тексту 20 пікселів
<br>

Відстань до тексту 1 піксель
</body>

```

Відповідне вікно браузера показано на рис. 2.20.

12. Додамо HTML-код для визначення альтернативного зображення:

```

...
<h2>Використання мініатюр</h2>

</body>

```




Рис. 2.20. Відокремлення тексту від рисунків за рахунок відступів

13. Розглянути можливість використання на сайтах відео-зображень. Для цього необхідно в папку HTML вставити файл з відео, наприклад **CLOCK.AVI** та додати в файл picture.html HTML-код:

```
...  
<h2>Використання відеозображень</h2>  
  
</body>
```

Моментальний знімок перегляду відео з файлу показано на рис. 2.21.

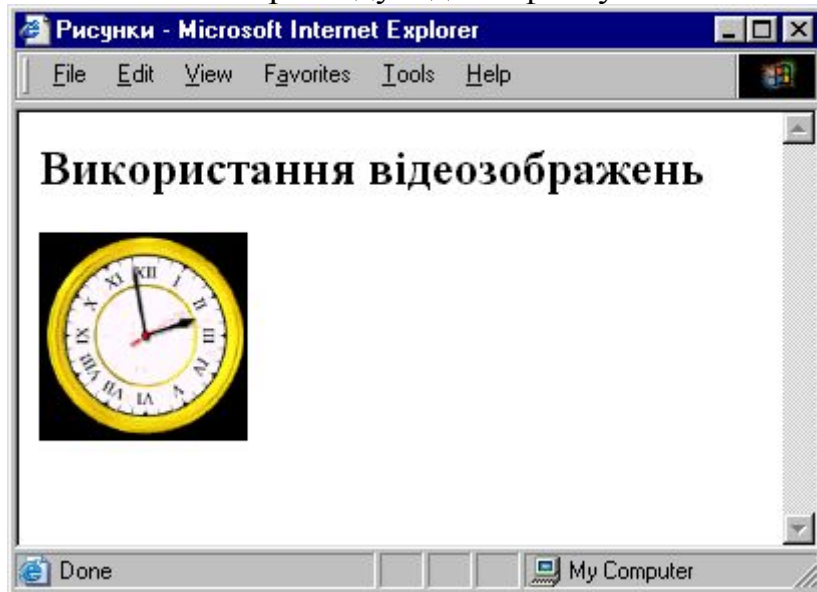


Рис. 2.21. Перегляд відео у вікні браузера



14. Модифікувати HTML-код для прокрутки відео-ролика **CLOCK.AVI** два рази:

```

```

Провести оновлення вікна браузера.

15. Модифікувати HTML-код для прокрутки відео ролика **CLOCK.AVI** два рази, при цьому показ починається після наведення миші на рисунок:

```

```

Провести оновлення вікна браузера.

16. Модифікувати HTML-код для того, щоб відео-ролик **CLOCK.AVI** прокручувався постійно:

```

```

Провести оновлення вікна браузера.

17. Розглянути можливість використання на сайтах звуку. Для цього необхідно в папку HTML вставити звуковий файл, наприклад **START.WAV** та додати в файл picture.html HTML-код:

...

```
<embed src="START.WAV" hidden="true" autostart="true" loop="True">
```

```
</body>
```

Провести оновлення вікна браузера.

Контрольні питання:

1. Як визначити розмір вертикального відступу від тексту до границі рисунку?
2. Як визначити розмір горизонтального відступу від тексту до границі рисунку?
3. В чому полягає специфіка плаваючого вирівнювання рисунків?
4. Навіщо потрібно записувати альтернативний текст при визначенні рисунків?
5. Навіщо потрібні мініатюри при визначенні рисунків?
6. Як вирівняти рисунок по верхньому краю рядка?
7. В чому полягає різниця між вирівнюванням рисунку bottom та absbottom?
8. В чому полягає різниця між вирівнюванням рисунку absmiddle та middle?
9. В чому полягає різниця між вирівнюванням рисунку texttop та top?
10. Як визначити товщину границі рисунка?



2.4. Лабораторна робота №4 "Створення гіперпосилань"

Мета роботи: навчитись створювати гіперпосилання.

Завдання:

Створити три теки з назвами А,В,С. В теці А створити HTML-документ 1.html, в теці В – 2.html, а в теці С – 3.html. В кожному із цих документів визначити гіперпосилання на два інших документи як з абсолютною, так із відносною адресацією. Крім того документ 1.html повинен завжди відкриватись в новому вікні браузеру.

Методичні вказівки.

1. Скопіювати в теку HTML видані викладачем графічні файли Eifel_small.jpg, Eifel.jpg, Flower.jpg.
2. В теці HTML створити текстовий документ з назвою Eifel.html, та записати в ньому такий HTML-код:

```
<html>  
<head><title>Ейфелева вежа</title></head>  
<body>  
      
</body>  
</html>
```

3. В теці HTML створити текстовий документ з назвою Link.html та записати в ньому такий HTML-код:

```
<html>  
<head><title>Створення гіперпосилань</title></head>  
<body>  
    У Франції є дуже багато визначних пам'яток.Спочатку ми  
    відвідаємо Ейфелеву вежу.  
</body>  
</html>
```

4. Визначити в документі Link.html текстове гіперпосилання на документ Eifel.html, який розміщений в одному каталозі з файлом Link.html. Для цього модифікувати HTML-код та переглянути документ (рис.2.22):

```
...  
    У Франції є дуже багато визначних пам'яток.  
    Спочатку ми відвідаємо <a href="Eifel.html">Ейфелеву вежу</a>  
</body>
```

Гіперпосилання буде завантажувати файл Eifel.html в те ж саме вікно браузера. Результат виконання гіперпосилання показаний на рис. 2.23.

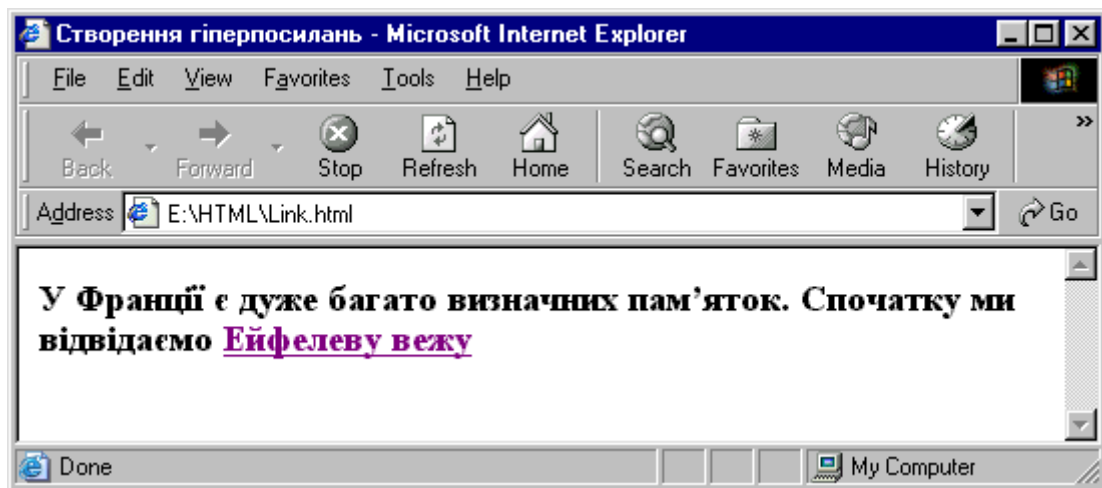


Рис. 2.22. Відображення текстового гіперпосилання у вікні браузера

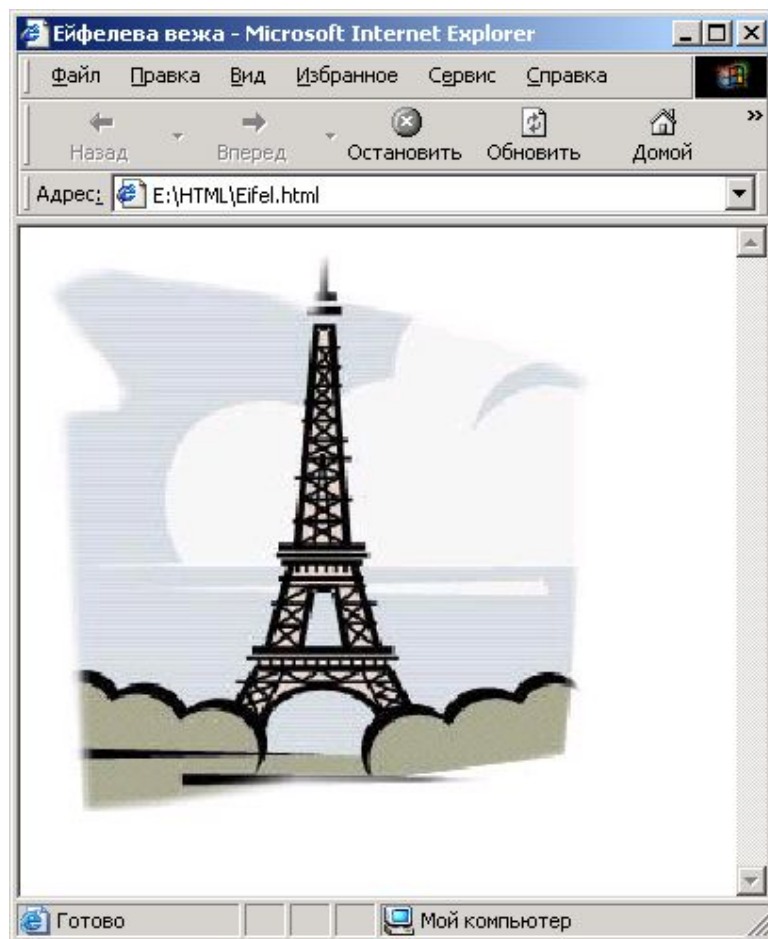


Рис. 2.23. Результат, отриманий при реалізації текстового гіперпосилання

5. Визначити в документі Link.html графічне гіперпосилання на документ Eifel.html. Додати для цього відповідний HTML-код та переглянути документ Link.html (рис.2.24):

```

...
<a href="Eifel.html"></a>
</body>

```

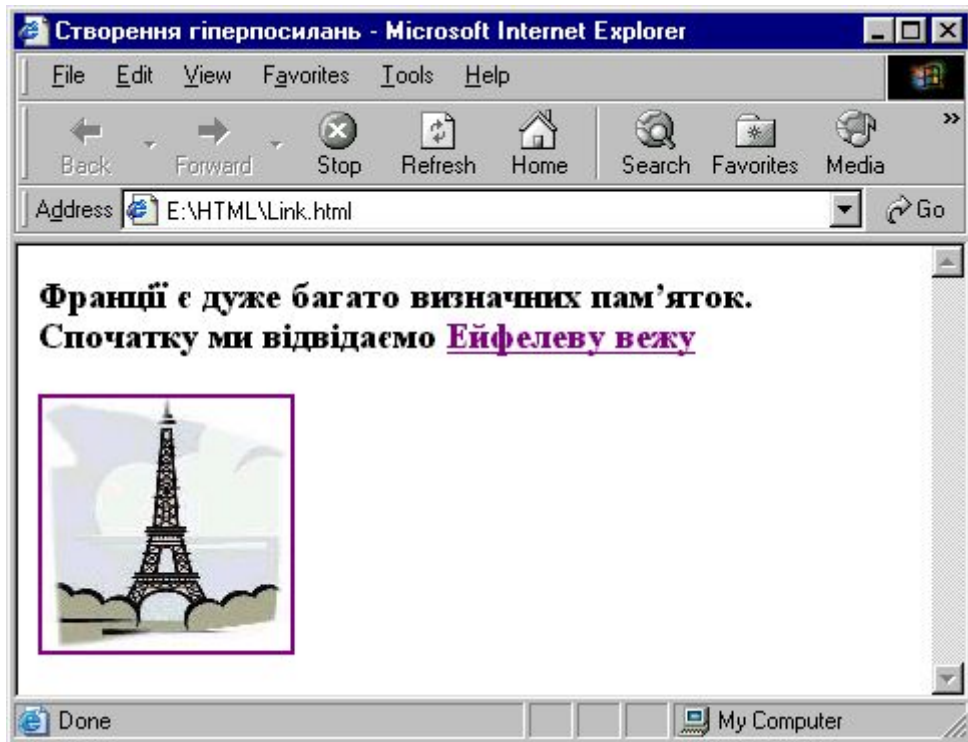


Рис. 2.24. Відображення графічного гіперпосилання у вікні браузера

Результат реалізації даного графічного гіперпосилання не повинен відрізнятися від результату тестового гіперпосилання, визначеного в п. 4 (рис. 2.23).

6. Використання відносної адресації в гіперпосиланнях.

6.1. Створити в теці HTML теку А. В теці А створити текстовий документ з назвою `Zamok.html`.

6.2. В файлі `Zamok.html` визначити наступний HTML-код:

```
<html>
  <head>
    <title>Замок</title>
  </head>
  <body>
    
  </body>
</html>
```

Переглянути документ `Zamok.html`, рис. 2.25

6.3. Визначити в документі `Link.html` текстове та графічне гіперпосилання на документ `Zamok.html`:

```
...
<a href="A/Zamok.html"><center>Замок</center></a>
<br><center><a href="A/Zamok.html"></a></center>
</body>
```

Переглянути оновлений документ `Link.html`, рис. 2.26.

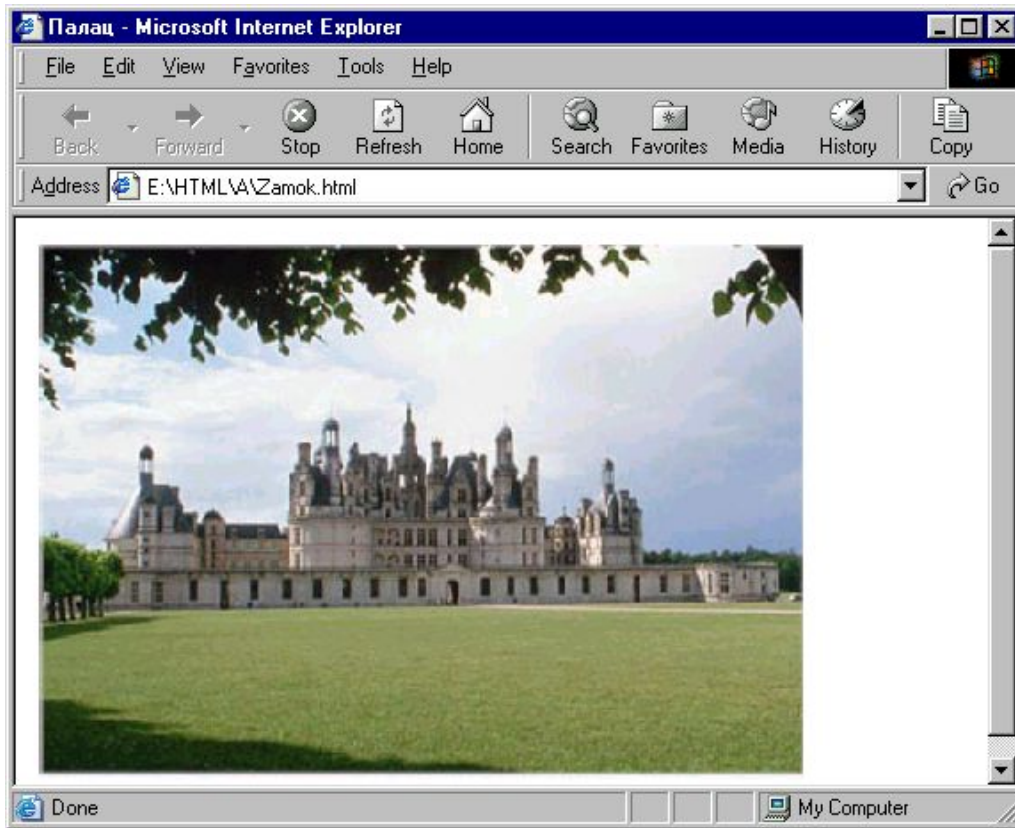


Рис. 2.25 HTML-документ Zamok.html

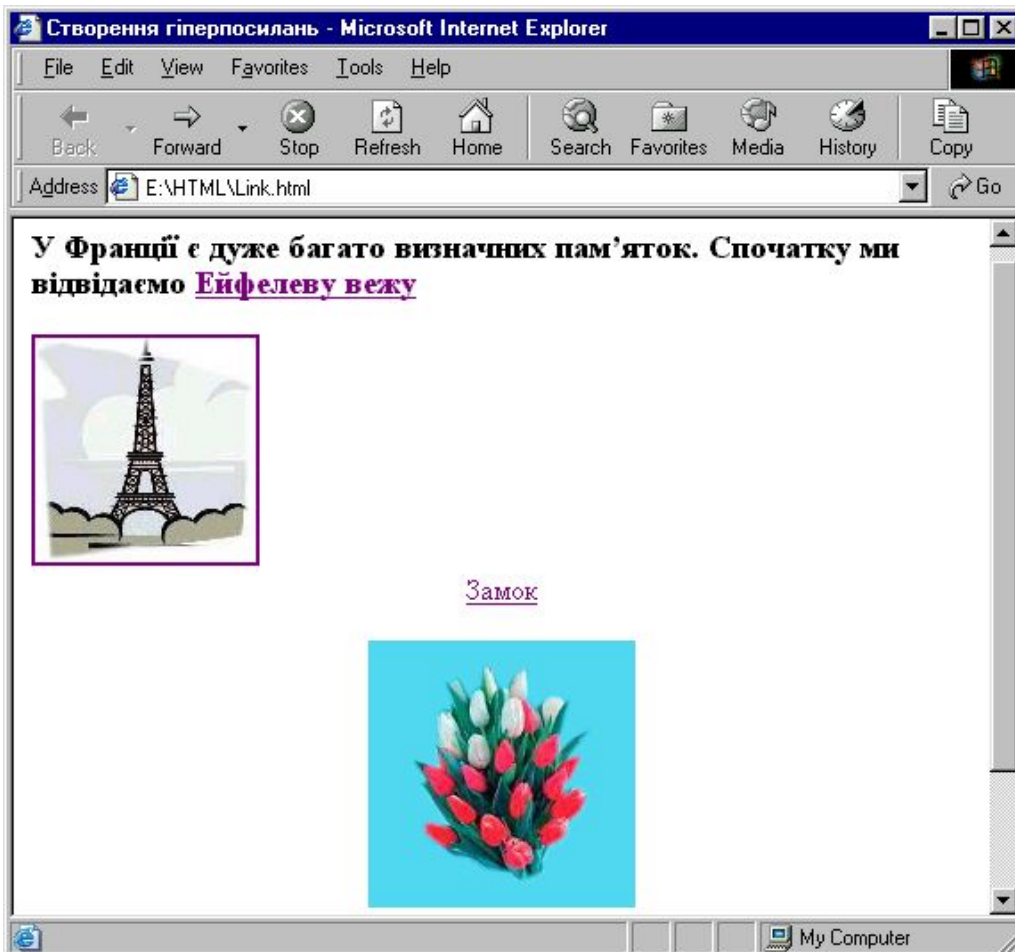


Рис. 2.26. Відображення оновленого документу Link.html

Реалізація визначених нами відносних посилань на документ `Zamok.html` показана на рис. 2.25

6.4. На одному рівні з текою HTML створити теку B. Наприклад, якщо тека HTML знаходиться в корені диску E, то і теку B створити в корені диску E. Скопіювати файл `Zamok.html` в теку B. В документі `Link.html` записати текстове гіперпосилання на документ `Zamok.html`, що знаходиться в теці B:

```
...  
<br><a href=" ../B/Zamok.html">На документ Zamok.html, що  
знаходиться в теці B</a>  
</body>
```

Реалізація цього посилання буде відрізнятися від представленого на рис. 2.25 тільки тим, що в адресному рядку браузера буде записано інший шлях - `E:\B\Zamok.html`.

7. Використання абсолютної адресації в гіперпосиланнях.

7.1. Використовуючи абсолютну адресацію додати в документі `Link.html` текстове гіперпосилання на HTML-документ `Zamok.html`, що знаходиться в теці B, тобто за адресою `E:\B\Zamok.html`:

```
...  
<br><a href=" "E://B/Zamok.html">На документ Zamok.html, що  
знаходиться в теці B</a>  
</body>
```

Реалізація цього посилання не повинна відрізнятися від реалізації посилання, визначеного в п. 6.4.

7.2. Визначити в документі `Link.html` текстове та графічне гіперпосилання на сайт, що знаходиться за адресою `http://narod.yandex.ru`:

```
...  
<br><a href="http://narod.yandex.ru">Замок</a>  
<br><a href="http://narod.yandex.ru"></a>  
</body>
```

Реалізація кожного з цих посилань показана на рис. 2.27.

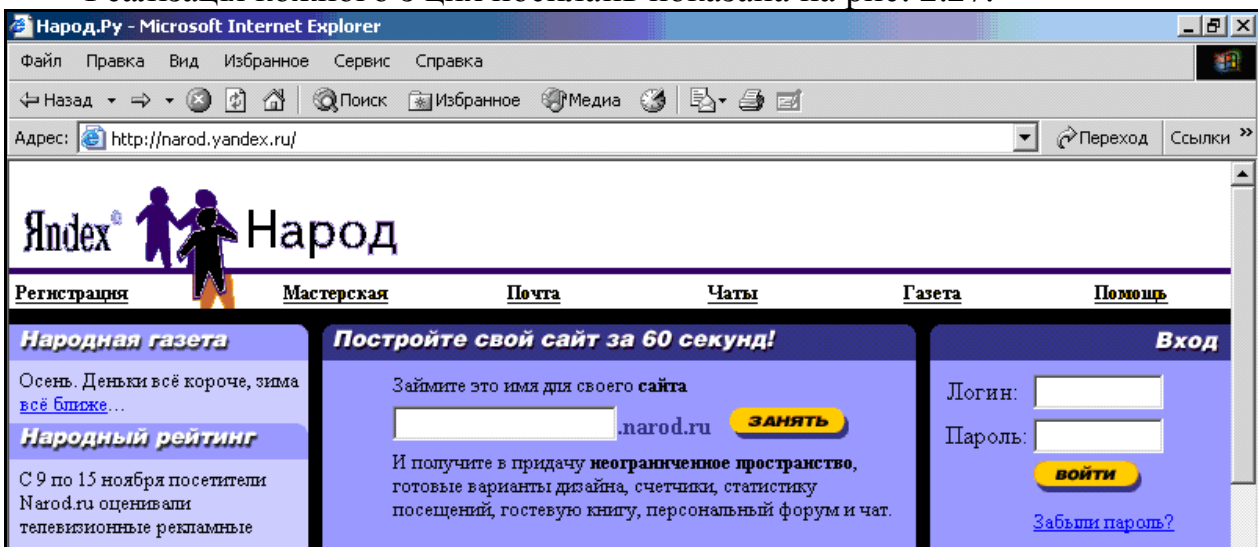


Рис. 2.27. Перехід по гіперпосиланню на сайт за адресою `http://narod.yandex.ru`



3. Використання гіперпосилань в середині HTML-документу.

8.1. В теці HTML створити текстовий документ з назвою `Recipe.html` та записати в ньому такий HTML-код:

```
<html>
<head><title>Створення внутрішнього гіперпосилання</title></head>
<body>
  <h1>Рецепти</h1>
  <h3>1. Кошки з кавовим кремом </h3>
  <h3>2. Тістечка "Наполеон"</h3>
  <h3>3. Яблука в заварному тісті </h3>
  <h2 align="center">1. Кошки з кавовим кремом </h2>
  <h3>Тісто:</h3>
  <p align="justify">300 г борошна, 100 г цукру, 150 г вершкового
масла або маргарину.</p>
  <h3>Крем:</h3>
  <p align="justify">1 ст. ложка борошна, 2 жовтки, 100 г цукру, 100
г вершкового масла, 3 ст. ложки настою міцної кави, 100 г посічених
горіхів.</p>
  ...
  <h2 align="center">3. Яблука в заварному тісті </h2>
  <h3>Тісто:</h3>
  <p align="justify">80 г розтопленого вершкового масла, 1 склянка води,
150 г борошна, 3 яєць, на кінець ножа</p>
  <h3>Начинка:</h3>
  <p align="justify">1 кг яблук, 1 ст. ложка варення.</p>
  <h3>Приготування </h3>
  <p align="justify">До гарячого розтопленого масла долити гарячої води і
закуп'ятити...</p>
</body>
</html>
```

8.2. Створити в документі `Recipe.html` посилання на текст третього пункту документу – "Яблука в заварному тісті". Для цього:

- Розмістити безпосередньо перед третім пунктом "якір" з ім'ям `apple`:

```
...
<a name= apple> </a><h2 align="center">3. Яблука в заварному тісті
</h2>
```

- Модифікувати HTML-код файлу `Recipe.html` для визначення самого внутрішнього посилання. При цьому в параметрі `href` вказати ім'я "якоря" з префіксом #:

```
...
<h3><a href="#apple">3. Яблука в заварному тісті</a> </h3>
...
```


Відображення документу Resepti.html відразу після завантаження та після реалізації визначеного нами внутрішнього посилання показані на рис. 2.28 та рис. 2.29.

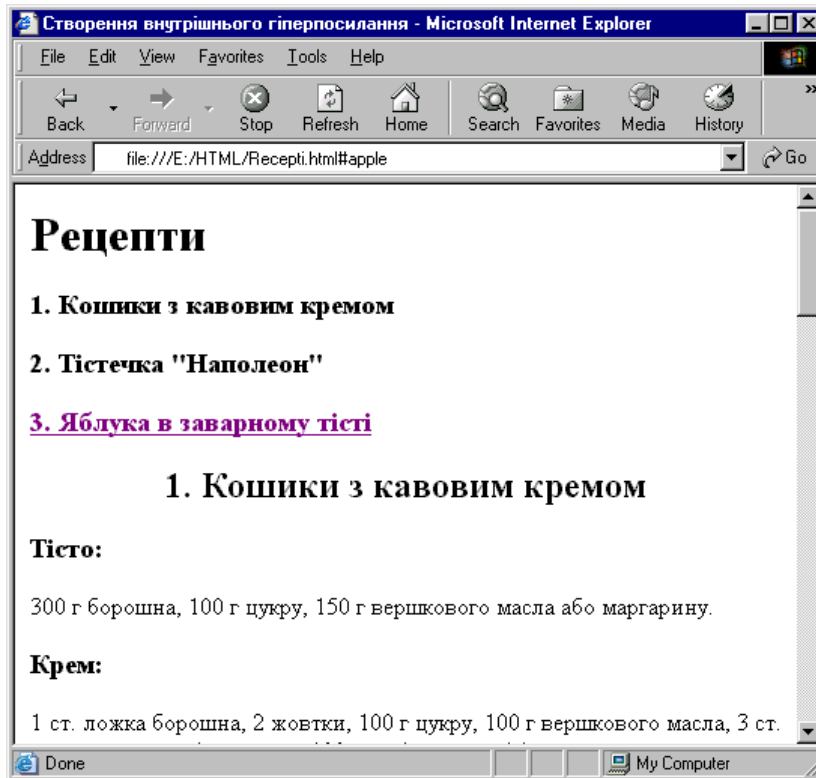


Рис. 2.28. Відображення документу Resepti.html відразу після завантаження

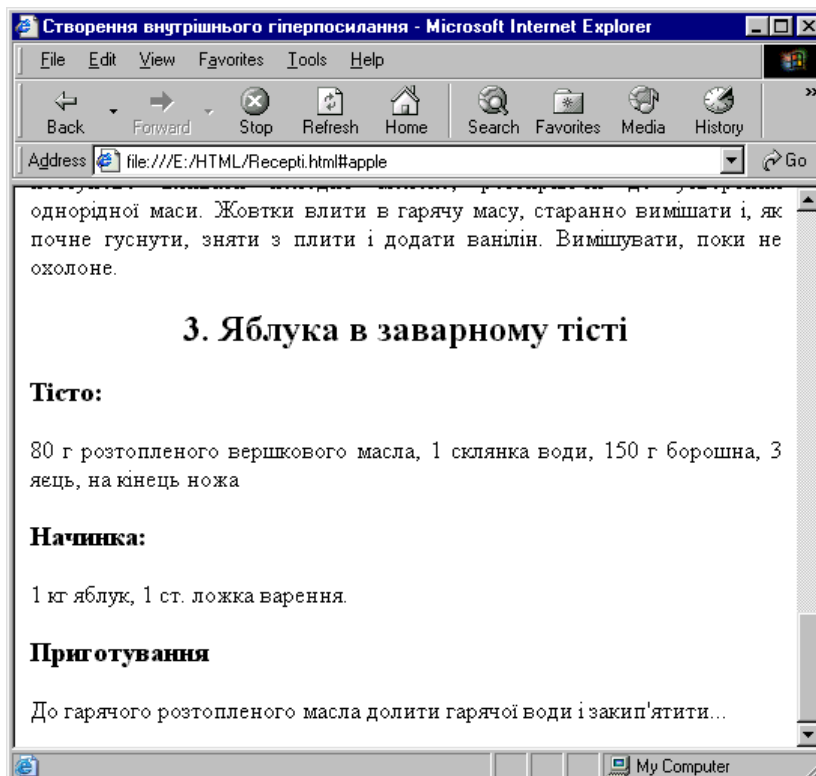


Рис. 2.29. Відображення документу Resepti.html після реалізації внутрішнього посилання

11. Визначити в документі Link.html текстове гіперпосилання на рисунок Flower.jpg, що знаходиться в папці HTML:

```
...  
<br><a href="Flower.jpg">Квіти</a>  
</body>
```

Реалізація даного посилання показана на рис. 2.30.

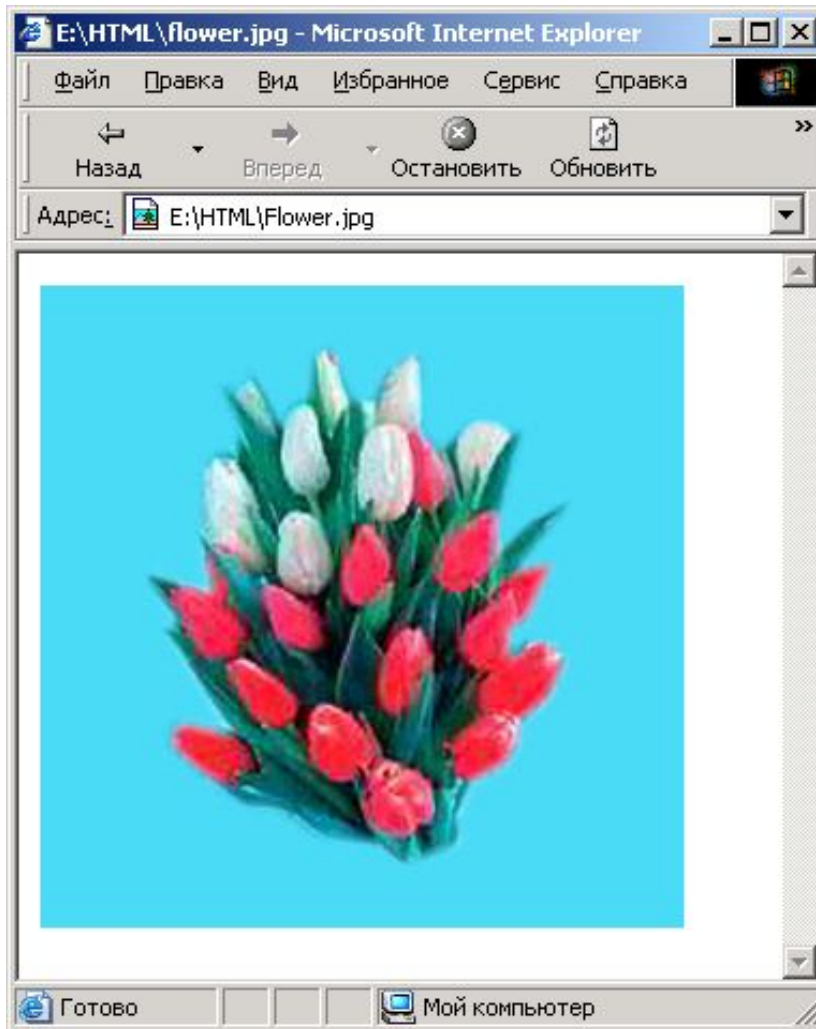


Рис. 2.30. Перехід по текстовому гіперпосиланню на рисунок Flower.jpg

12. Визначимо в документі Link.html текстове гіперпосилання на адресу електронної пошти:

```
...  
<br><a href="mailto:pochta@ukr.net ">Пошта</a>  
</body>
```

Реалізація даного посилання в випадку використання в якості поштового клієнту програми "Outlook Express" показана на рис. 2.31.

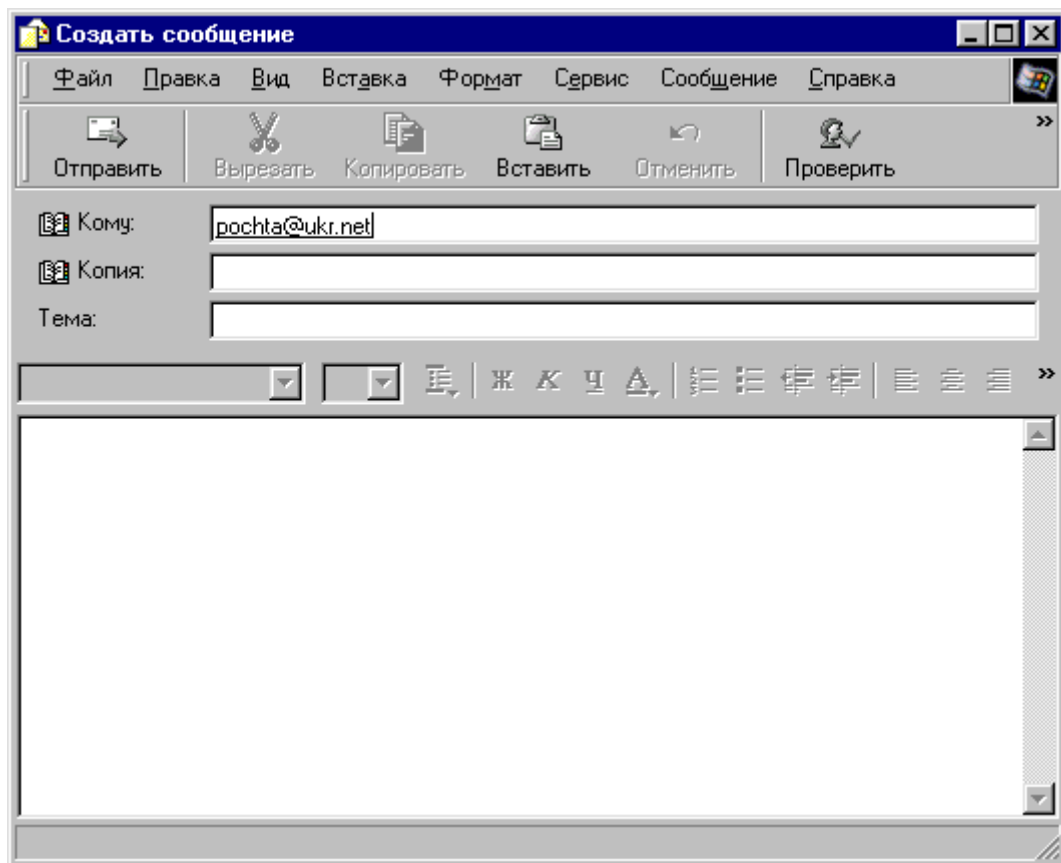


Рис. 2.31. Реалізація гіперпосилання на адресу електронної пошти pochta@ukr.net

Контрольні питання:

1. Як відкрити гіперпосилання в новому вікні браузера?
2. Як визначити якір в HTML-документі?
3. Як використовується в гіперпосиланнях абсолютна адресація?
4. Як використовується в гіперпосиланнях відносна адресація?
5. Як відкрити гіперпосилання в тому ж самому вікні браузера?
6. Правила запису текстових гіперпосилань.
7. Правила запису графічних гіперпосилань.
8. Як визначити гіперпосилання на рисунок?
9. Як визначити гіперпосилання в середині HTML-документу?
10. Як записати гіперпосилання на адресу електронної пошти?



2.5. Лабораторна робота №5 "Створення списків"

Мета роботи: опанувати основні прийоми роботи зі створення списків.

Завдання:

Визначити список відповідно зразка варіанта, показаного нижче.

Методичні вказівки.

1. Скопіювати в теку HTML виданий викладачем графічний файл star.jpg.
2. В теці HTML створити текстовий документ з назвою List.html, відкрити його за допомогою браузера та перейти до редагування HTML-коду
3. Створити маркірований список, елементами якого будуть місяці року. Для цього в файлі List.html записати HTML-код:

```
<html>
<head>
  <title>Створення списків</title>
</head>
<body>
  <ul>
    <b>Місяці року:</b>
    <li>Січень
    <li>Лютий
    <li>Березень
    <li>Квітень
    <li>Травень
    <li>Червень
    <li>Липень
    <li>Серпень
    <li>Вересень
    <li>Жовтень
    <li>Листопад
    <li>Грудень
  </ul>
</body>
</html>
```

Перегляд списку в браузері повинен відповідати рис. 2.32.



Рис. 2.32. Відображення браузером маркірованого списку

4. Розглянути можливість зміни типу маркеру. Необхідно відобразити маркери заповненими квадратами для перших чотирьох елементів списку, відобразити маркери незаповненими колами для других чотирьох елементів списку та відобразити маркери заповненими колами для останніх чотирьох елементів списку. Реалізувати зміну типу маркерів за допомогою параметра *type* тега ``. Для цього модифікувати код списку:

```

...
<ul>
  <b>Місяці року:</b>
  <li type="square">Січень
  <li type="square">Лютий
  <li type="square">Березень
  <li type="square">Квітень
  <li type="circle">Травень
  <li type="circle">Червень
  <li type="circle">Липень

```



```
<li type="circle">Серпень  
<li type="disk">Вересень  
<li type="disk">Жовтень  
<li type="disk">Листопад  
<li type="disk">Грудень  
</ul>  
</body>
```

Переглянути цей маркірований список у браузері (рис. 2.33)



Рис. 2.33. Відображення маркірованого списку, для елементів якого визначено різні значення параметра type

5. Створити список з графічними маркерами. При цьому можна обійтись без тегів ``. Перед кожним елементом списку необхідно вставити графічне зображення. Тоді елементи списку відокремляти один від одного за допомогою тегу примусового переведення строки `
`. Графічний файл `star.jpg` повинен знаходитись в тому ж самому каталозі, що і наш файл `List.jpg`, тобто в папці HTML на диску E. Записати HTML-код:



```
...  
<ul>  
  <b>Місяці року:</b><br><br>  
  Січень<br>  
  Лютий<br>  
  Березень<br>  
  Квітень<br>  
  Травень<br>  
  Червень<br>  
  Липень<br>  
  Серпень<br>  
  Вересень<br>  
  Жовтень<br>  
  Листопад<br>  
  Грудень  
</ul>  
</body>
```

Переглянути цей маркірований список у браузері (рис. 2.34)

6. Створити нумерований список. Реалізувати це за допомогою тегу - контейнеру `` ``. Елементами нумерованого списку будуть знову місяці року. Записати відповідний HTML-код та переглянути цей нумерований список у браузері (рис. 2.35) :

```
...  
<ol>  
  <b>Перший квартал включає наступні місяці року:</b>  
  <li>Січень  
  <li>Лютий  
  <li>Березень  
</ol>  
</body>
```

6. Визначимо різні види нумерації списку. Реалізуємо це за допомогою параметру `type` тегу ``. Записуємо HTML-код для вказаного списку та переглянемо його у браузері (рис. 2.36):

```
...  
<ol>  
  <b>Квартали включають наступні місяці року:</b>  
  <li type="I">Січень, лютий, березень  
  <li type="i">Квітень, травень, червень  
  <li type="A">Липень, серпень, вересень  
  <li type="a">Жовтень, листопад, грудень  
</ol>  
</body>
```

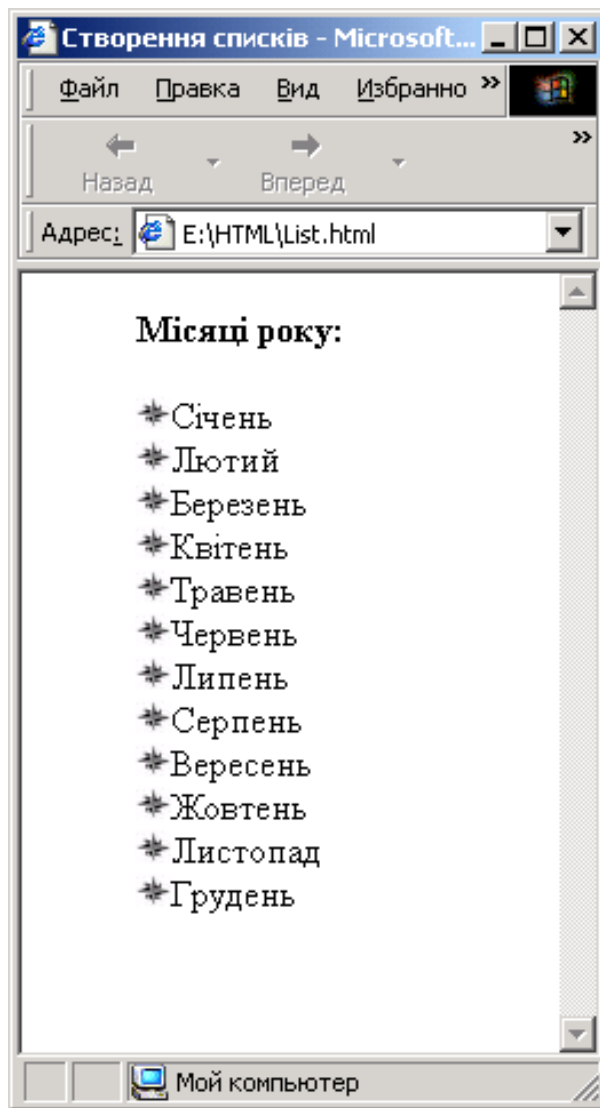


Рис. 2.34. Відображення маркірованого списку з графічними маркерами

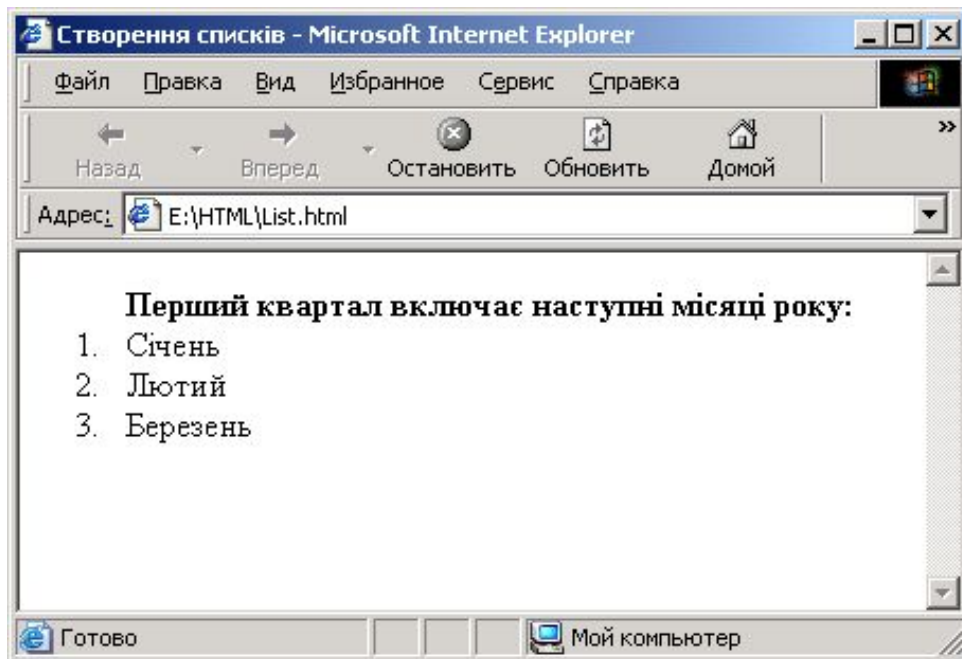


Рис. 2.35 Відображення нумерованого списку

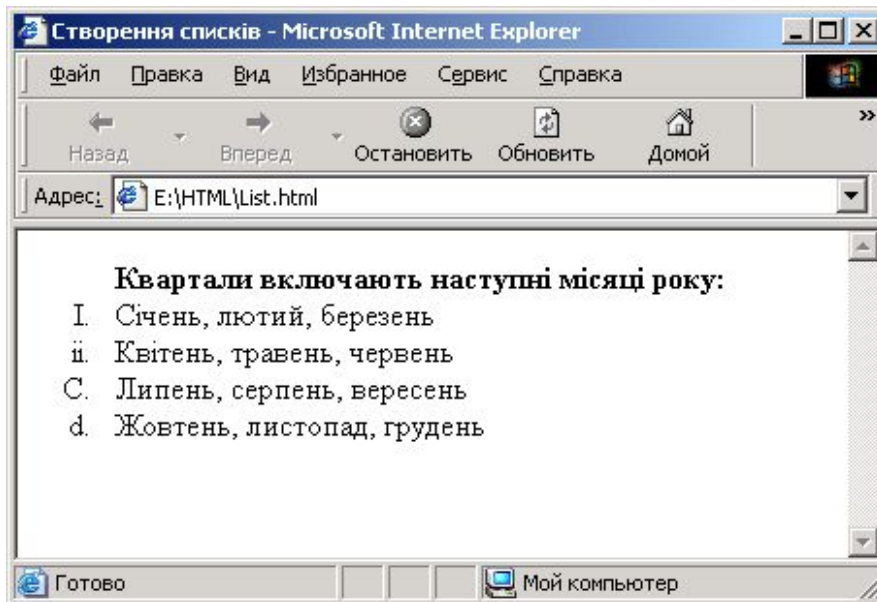


Рис. 2.36. Відображення нумерованого списку, для якого визначено різні види нумерації списку

7. Визначимо нумерований список, який складається з п'яти елементів. В цьому списку нумерація починається з 1995, та в третьому рядку списку початок нумерації змінюється на 2003. Записуємо відповідний HTML-код :

```

...
<ol start="1995">
  <li>
  <li>
  <li value="2003">
  <li>
  <li>
</ol>
</body>

```

Переглянути цей нумерований список у браузері (рис. 2.37)

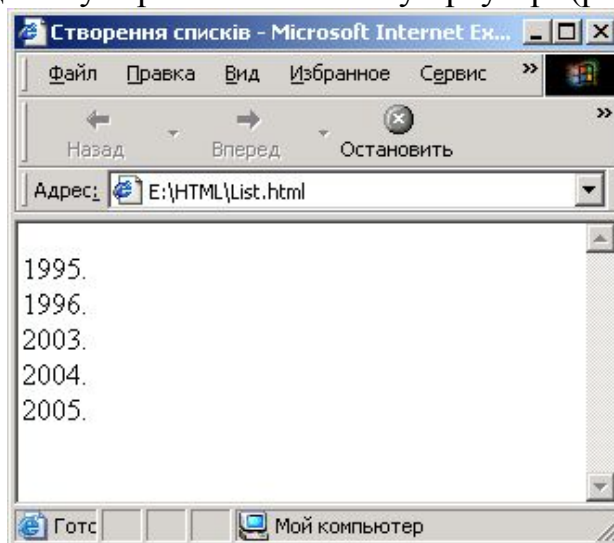
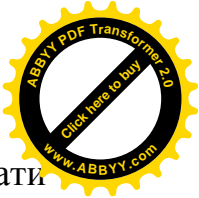


Рис. 2.37. Відображення нумерованого списку, для якого визначено заміну початку нумерації списку та заміну нумерації пунктів в середині списку



3. Створити список визначень. В першій частині пункту списку записати термін, який визначає темперамент людини. Реалізувати це за допомогою тегу `<dt>`. А в другій частині пункту списку записати пояснення, яке розкриває значення терміну. Це реалізувати за допомогою тегу `<dd>`. Записати HTML-код для такого списку визначень :

```
...
<dl>
  <h3 align="center"> Класифікація типових темпераментів людини, <br>
яка заснована на поглядах Гіппократа </h3>
  <dt> Флегматик
  <dd> Пасивний, дуже працездатний, повільно пристосовується; <br>
настрій стійкий, мало піддається зовнішньому впливу; <br> млявість
емоційних реакцій та повільність у вольовій діяльності <br><br>
  <dt> Сангвінік
  <dd> Активний, енергійний, легко пристосовується; <br> живість та
рухливість емоційних реакцій, швидкість та сила вольових проявів <br><br>
  <dt> Холерик
  <dd> Активний, дуже енергійний, наполегливий; <br> поривчастість та
сила емоційних реакцій, бурні вольові прояви <br> <br>
  <dt> Меланхолік
  <dd> Пасивний, легко стомлюється, важко пристосовується; <br>
слабкість вольових проявів та перевага подавленого настрою, невпевненість в
собі
</dl>
</body>
```

Переглянути цей список визначень у браузері (рис. 2.37)

9. Створимо вкладений список, який показує супутники деяких планет. Розглянемо випадок, коли в кожен елемент маркірованого списку вкладений свій нумерований список. Записуємо відповідний HTML-код:

```
...
<ul>
  <b>Супутники деяких планет</b>
  <li>Земля
  <ol>
    <li>Луна
  </ol>
  <li>Марс
  <ol>
    <li>Фобос
    <li>Деймос
  </ol>
  <li>Уран
  <ol>
    <li>Аріель
```

```

<li>Умбрі ель
<li>Тітанія
<li>Оберон
<li>Міранда
</ol>
<li>Нептун
<ol>
<li>Тритон
<li>Нереїда
</ol>
</ul>
</body>

```

Переглянемо цей вкладений список у браузері (рис. 2.39)

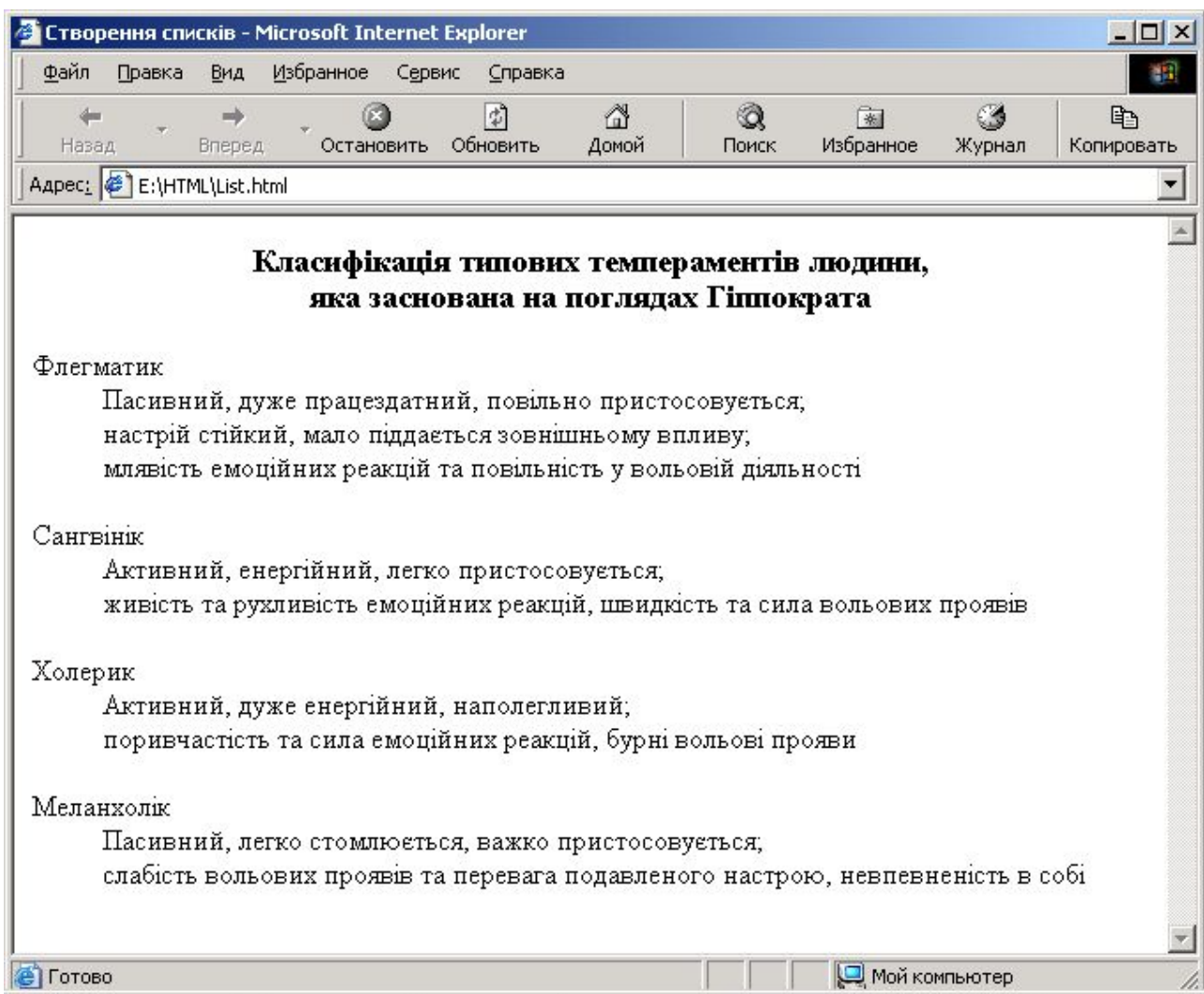


Рис. 2.38. Відображення списку визначень, який розкриває класифікацію темпераментів людини

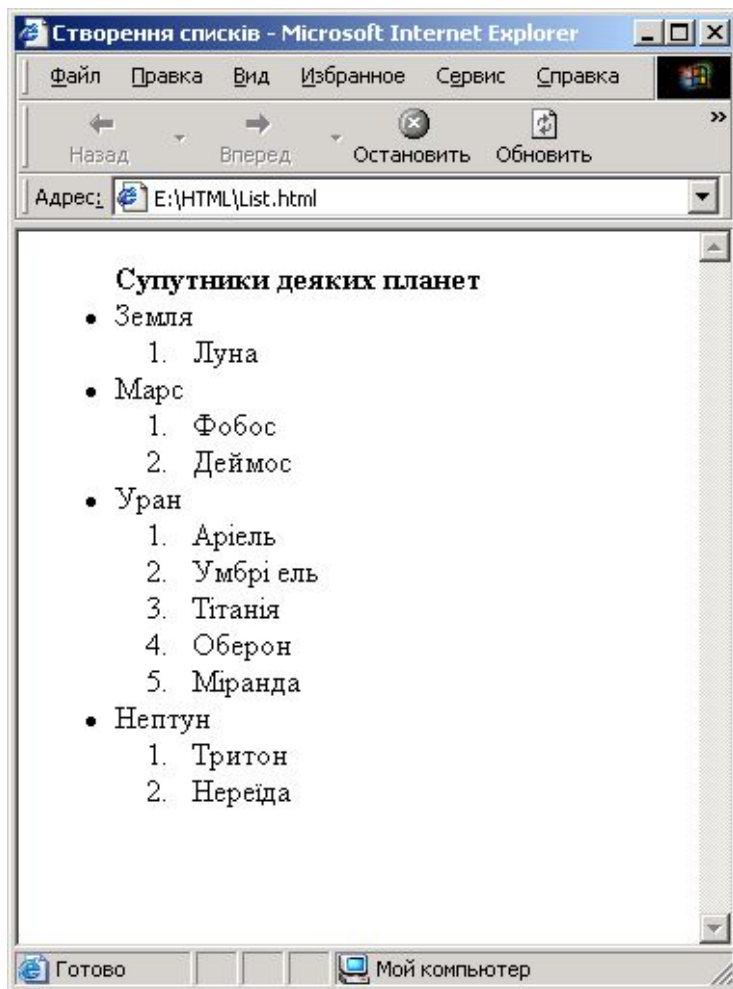


Рис. 2.39. Відображення вкладеного списку, який показує супутники деяких планет

Варіанти завдань до лабораторної роботи № 5

Варіант № 1

Електротехнічний факультет

1. Електротехніка
 - ЕС
 - ЕСІМ
 - ЕСЕ
2. Електромеханіка
 - ЕМО
 - ЕАПУ



Варіант № 2

Види транспорту

- 1) Сухопутний
 - Залізничний
 - Автомобільний
- 2) Водний
 - Морський
 - Річковий
- 3) Повітряний
 - Авіаційний

Варіант № 3

Природні ресурси

- Вичерпні
 1. Біологічні
 2. Земельні
 3. Мінеральні
- Невичерпні
 1. Водні
 2. Кліматичні

Варіант № 4

Нематеріальні послуги

- ✓ Необхідні
 - 1) Освіта
 - 2) Культура
 - 3) Охорона здоров'я
 - 4) Фізична культура
- ✓ Вимушено необхідні
 - 1) Державне управління
 - 2) Оборона
 - 3) Охорона суспільного порядку

Варіант № 5

Виторг реалізації продукції

- 1) Витрати на виробництво продукції
- 2) Прибуток підприємства
 - Податки
 - Виконання зобов'язань перед кредиторами
 - Прибуток, що залишається в розпорядженні підприємства



Варіант № 6

Виробництво продукції

1. Засоби виробництва
 - Засоби праці
 - Предмети праці
2. Виробничі ресурси
 - Трудові ресурси
 - Інформаційні ресурси
 - Природні ресурси

Варіант № 7

Виробничий цех

- ✧ Цукеркове виробництво
 1. Лінія “Помада
 2. Лінія “Ірис
- ✧ Бісквітне виробництво
 1. Лінія “Десерт1”
 2. Лінія “Десерт2”
 3. Лінія “Рулети
- ✧ Шоколадне виробництво
 1. Лінія “Плитки
 2. Лінія “Шоколадні цукерки ”

Варіант № 8

Матеріальні послуги

- Ті, що матеріалізуються в предметах свого діяння
 - 1) Торгівля
 - 2) Громадське харчування
 - 3) Житлово-комунальні
 - 4) Побутове обслуговування
- Ті, що не матеріалізуються в предметах свого діяння
 - 1) Транспорт
 - 2) Зв'язок



Варіант № 9

Вигоди від будівництва моста

- I. Економічні вигоди
 - ✓ Час
 - ✓ Доход
 - ✓ Торгівля в околицях
 - ✓ Торгівля уздовж
 - ✓ Роботи по будівництву
- II. Соціальні вигоди
 - ✓ Безпека і надійність
 - ✓ Зв'язки
 - ✓ Гордість місцевих жителів
- III. Вигоди середовища
 - ✓ Комфорт
 - ✓ Доступність
 - ✓ Естетика

Варіант № 10

Витрати на будівництво моста

- ✧ Економічні витрати
 - I. Капітальні укладання
 - II. Управління і експлуатація
 - III. Завершення паромного бізнесу
- ✧ Соціальні витрати
 - I. Зміна стилю життя
 - II. Роз'єднання людей
 - III. Розміщення людей
- ✧ Витрати середовища
 - I. Підвищення шкідливих викидів
 - II. Забруднення води через міст
 - III. Порушення екології

Варіант № 11

ОБМЕЖЕННЯ НА ЗДІЙСНЕННЯ ПРИЗНАЧЕННЯ

- A. Внутрішні обмеження
 - Сприйняття
 - Унікальні проблеми
 - Конфлікт
 - Інерція
- B. Обмеження середовища
 - Динаміка і планування
 - Турбулентність
 - Зворотний зв'язок із запізнюванням



Варіант № 12

Життєвий цикл технічного об'єкта

- 1) Розробка ТЗ
- 2) Проектування
 - Попереднє
 - Ескізне
 - Технічне
 - Робоче
- 3) Експериментальний зразок
- 4) Серійне виробництво
- 5) Експлуатація

Варіант № 13

Системи планування рішення

- a). SPADE
 1. Постановка задачі
 2. Планування рішення
 3. Виконання розрахунків
- b). ПРИЗ
 4. Формування опису задачі
 5. Планування обчислень
 6. Реалізація
- c). АФП
 7. Формульне проектування методом бісекцій

Варіант № 14

Навчальне навантаження

- Навчальні завдання
 - a). Лекції
 - b). Лабораторні заняття
 - c). Практичні заняття
 - d). Семінарські заняття
- Самостійна робота
 - a). Опрацювання лекційного матеріалу
 - b). Підготовка до лабораторних робіт
 - c). Підготовка до практичних занять
 - d). Підготовка до семінарів
 - e). Виконання індивідуальних завдань
 - f). Самостійне опрацювання розділів навчальної програми
 - g). Підготовка до мольного контролю



Варіант № 15

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК І ТЕХНОЛОГІЙ

- I. Напрямок підготовки "Економічна кібернетика"
 - ЕКІ
- II. Напрямок підготовки "Комп'ютерні науки"
 - ІУС
 - КЕМ
 - ПКД
- III. Напрямок підготовки "Комп'ютерна інженерія"
 - КСД
 - КС
 - СП
- IV. Напрямок підготовки "Програмна інженерія"
 - ПС

Варіант № 16

КЛАСИЧНА ПОЛІТИЧНА ЕКОНОМІЯ

- a). Консервативний напрямок
 - Фашистські теорії
 - Корпоративізм
 - Неоконсерватизм
- b). Класичний ліберальний напрямок
 - Неоінституціоналізм
 - Нова класична економіка
 - Монетаризм
- c). Сучасний ліберальний напрямок
 - Неокейнсіансько-неокласичний синтез
 - Економіка добробуту
 - Посткейнсіанство
- d). Радикальний напрямок
 - Традиційний інституціоналізм
 - Соціал-демократичні теорії
 - Теорія ортодоксального марксизму

Варіант № 17

КУПІВЛЯ ЦІННИХ ПАПЕРІВ

- В емітента
 - A. Акціонерного товариства
 - B. Банку
 - C. Держави
- У торговця цінними паперами
- Через мережу УЦСА
- На позабіржовому ринку
- У приватної особи



Варіант № 18

ВИГОДИ РОЗМІЩЕННЯ ПІДПРИЄМСТВ ЗА КОРДОНОМ

- ✓ Економічні вигоди
 - 1) Дешева робоча сила
 - 2) Фінансова допомога країни-господаря
 - 3) Близькість до ринку
 - 4) Сила валюти країни-господаря
- ✓ Управлінські вигоди
 - 1) Знання місцевих умов ринку
 - 2) Неістотне втручання держави
 - 3) Надійність транспортних засобів

Варіант № 19

ВИТРАТИ РОЗМІЩЕННЯ ПІДПРИЄМСТВ ЗА КОРДОНОМ

- A. Економічні витрати
 - Висока вартість сировини
 - Великі місцеві податки на вироблені товари
 - Високі тарифи на матеріали, що імпортуються
- B. Управлінські витрати
 - Слабка підготовка персоналу
 - Політична нестабільність уряду країни господаря
 - Участь місцевих управлінців
 - Мовні і культурні бар'єри

Варіант № 20

МОРФОЛОГИЧЕСКИЙ СИНТЕЗ

- ✧ Морфологічний пошук
 - 1. Фільтрація
 - 2. Зондування
 - 3. Вдосконалення прототипу
- ✧ Морфологічне конструювання

Варіант № 21

РОЗРАХУНКОВЕ ОБСЛУГОВУВАННЯ ПЛАСТИКОВИХ КАРТОК

- ВИДАЧА І РІЧНЕ ОБСЛУГОВУВАННЯ
 - a). Оформлення основної картки
 - b). Розрахункове обслуговування основної картки
 - c). Оформлення додаткової картки
- ДОДАТКОВІ ПОСЛУГИ
 - a). Поповнення рахунку готівкою
 - b). Поповнення рахунку безготівковими засобами
- КОМІСІЙНІ ОПЕРАЦІЇ
 - a). У національній валюті
 - b). В іноземній валюті



Варіант № 22

ВИДИ ЦІННИХ ПАПЕРІВ

1. Акції
 - ✓ Прості
 - ✓ Привілейовані
 - ✓ Іменні
 - ✓ На пред'явника
2. Облігації державних і місцевих позик
3. Облігації підприємств
4. Казначейські зобов'язання держави
5. Ощадні сертифікати
 - ✓ Термінові чи до запитання
 - ✓ Іменні чи на пред'явника
6. Векселя
 - ✓ Прості
 - ✓ Перекладні
7. Приватизаційні цінні папери

Варіант № 23

ГОРИ РОЗРІЗНЯЮТЬСЯ:

- ▶ По висоті
 - 1) Низькі
 - 2) Середні
 - 3) Високі
- ▶ За віком
 - 1) Старі
 - 2) молоді
- ▶ За походженням
 - 1) Складчасті
 - 2) Брилеві

Варіант № 24

НОВА АКЦІЯ

- Кредитування юридичних і фізичних осіб під заставу
 - I. Від 30% і нижче в гривні
 - II. Від 18% до 16% у доларах США
- Прийом внесків від фізичних і юридичних осіб
 - I. До 17% річних у гривні
 - II. До 10% річних у доларах США, Євро
- Купівля-продаж безготівкової валюти на Міжбанківській валютній біржі (комісія)
 - I. Від 0,1%
 - II. До 0,45%



Варіант № 25

ПОСЛУГИ БАНКУ НАСЕЛЕННЮ

- A. Прийом платежів від фізичних і юридичних осіб
- B. Обслуговування платіжних карток
- C. Оренда індивідуальних депозитних осередків
- D. Внески для населення
 - У національній валюті
 - В іноземній валюті
- E. Розрахунково-касове обслуговування

Варіант № 26

ГІРСЬКІ ПОРОДИ

- ▲ Магматичні
 - a). Граніт
 - b). Базальт
- ▲ Осадові
 - a). Уламкового походження
 - b). Органічного походження
 - c). Хімічного походження
- ▲ Метаморфічні

Варіант № 27

ГЕОГРАФІЧНІ КАРТИ РОЗРІЗНЯЮТЬСЯ:

- I. По обхвату території
 - ▶ Карти світу і півкуль
 - ▶ Карти материків
 - ▶ Карти країн
- II. За змістом
 - ▶ Загальногеографічні карти
 - ▶ Тематичні карти
- III. По масштабу
 - ▶ Великомасштабні
 - ▶ Середньомасштабні
 - ▶ Дрібномасштабні

Варіант № 28

Розробка комп'ютерної системи

- a). Планування процесу
 - Ухвалення структури і методу виготовлення системи
 - Розрахунок вимог, витрат, часу на розробку
 - Уточнення плану супутніх заходів
- b). Розробка графіку робіт
- c). Створення моделі системи



Варіант № 29

Рівнини розрізняються:

- По рельєфу
 1. Плоскі
 2. Горбисті
- По висоті
 1. Низовини
 2. Піднесеності
 3. Плоскогір'я
- За походженням
 1. Первинні рівнини
 2. Вторинні рівнини

Варіант № 30

Історія розвитку Землі

- A. Кайнозойська ера
 - ▲ Четверічний період
 - ▲ Неогеновий період
 - ▲ Палеогеновий період
- B. Мезозойська ера
 - ▲ Крейдяний період
 - ▲ Юрський період
 - ▲ Тріасовий період
- C. Палеозойська ера
 - ▲ Пермський період
 - ▲ Кам'яновугільний період
 - ▲ Девонський період
 - ▲ Силурійський період
 - ▲ Ордовікський період
 - ▲ Кембрійський період
- D. Докембрійська ера
 - ▲ Протерозойський період
 - ▲ Архейський період

Контрольні питання:

1. Як визначається нумерований список?
2. Як визначається маркірований список?
3. Як визначається список визначень?
4. Як змінити тип маркеру в маркірованому списку?
5. Як змінити тип порядок нумерації в нумерованому списку?
6. Як визначити список з маркерами у вигляді рисунків?
7. Як почати нумерований список з довільного номеру?
8. Як визначаються вкладені списки?
9. Навіщо використовуються списки?
10. Як пронумерувати список за допомогою прописних англійських літер?



2.6. Лабораторна робота №6 "Створення таблиць "

Мета роботи: навчитись створювати таблиці.

Завдання:

В HTML-документі визначити таблицю, зміст та форматування якої відповідають таблиці свого варіанту.

Методичні вказівки.

1. Скопіювати в теку HTML видані викладачем графічні файли Zamok.jpg та Tochka.jpg, Yellow.jpg.
2. В теці HTML створити текстовий документ з назвою Table.html, відкрити його за допомогою браузера та перейти до редагування HTML-коду.
3. Створити таблицю, яка складається із двох рядків та трьох стовпців та містить текстову інформацію. Структура таблиці показана на рис. 2.40.

11	12	13
21	22	23

Рис. 2.40. Таблиця, яку необхідно відобразити у вікні браузера

Записати відповідний HTML-код файлу Table.html з цієї таблицею (границі поки що невидимі) та переглянути її у браузері (рис. 2.41):

```
<html>
<head>
  <title>Створення таблиць </title>
</head>
<body>
  <table>
    <tr>
      <td>11</td>
      <td>12</td>
      <td>13</td>
    </tr>
    <tr>
      <td>21</td>
      <td>22</td>
      <td>23</td>
    </tr>
  </table>
</body>
</html>
```

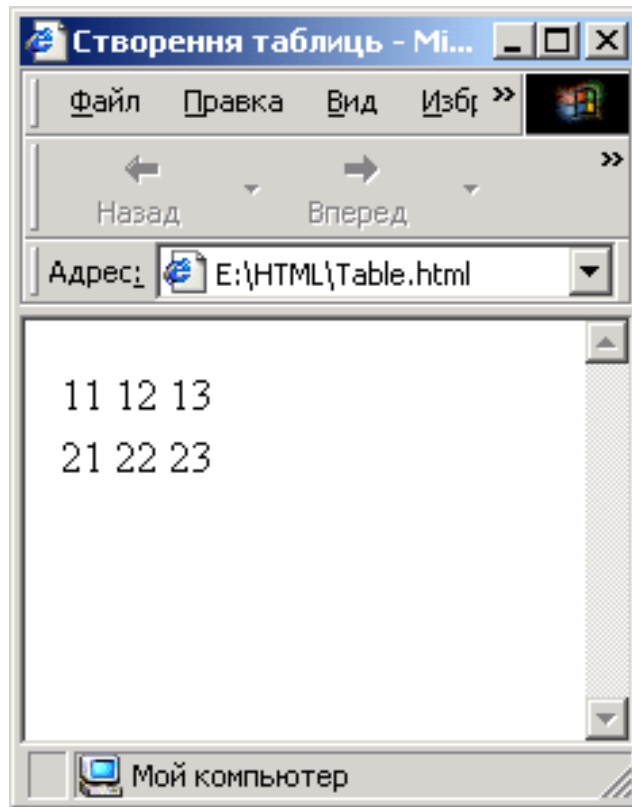


Рис. 2.41. Відображення таблиці, що складається із двох рядків та трьох стовпців у вікні браузера

4. Визначити границі таблиці. Реалізувати це за допомогою параметра `border`. Колір границі задається параметром `bordercolor="колір_границі"`. Для визначення таблиці із границями чорного кольору, товщина яких дорівнює 3 пікселі, необхідно записати наступний код:

```

...
<table border="3" bordercolor="#000000" >
  <tr>
    <td>11</td>
    <td>12</td>
    <td>13</td>
  </tr>
  <tr>
    <td>21</td>
    <td>22</td>
    <td>23</td>
  </tr>
</table>
</body>

```

Переглянути цю таблицю у браузері (рис. 2.42)

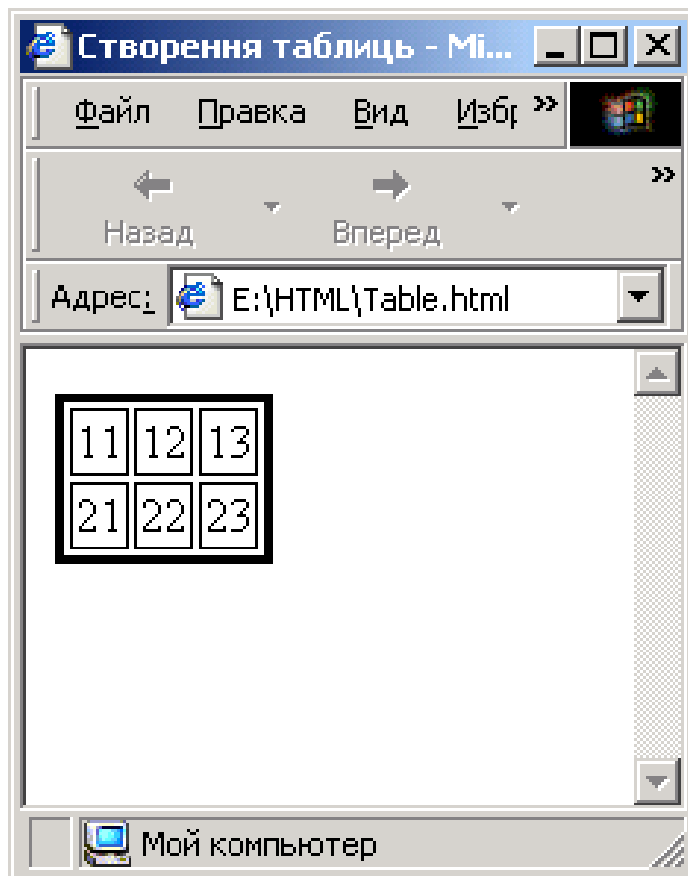


Рис. 2.42. Відображення таблиці із границями товщиною 3 пікселі у вікні браузера

5. Визначити фон таблиці. Реалізувати за допомогою параметру bgcolor = "колір_фону". Записати HTML-код для таблиці із сірим кольором фону та переглянути її у браузері (рис. 2.43):

```
...  
<table border="3" bordercolor="#000000" bgcolor="#999999">  
  <tr>  
    <td>11</td>  
    <td>12</td>  
    <td>13</td>  
  </tr>  
  <tr>  
    <td>21</td>  
    <td>22</td>  
    <td>23</td>  
  </tr>  
</table>  
</body>
```

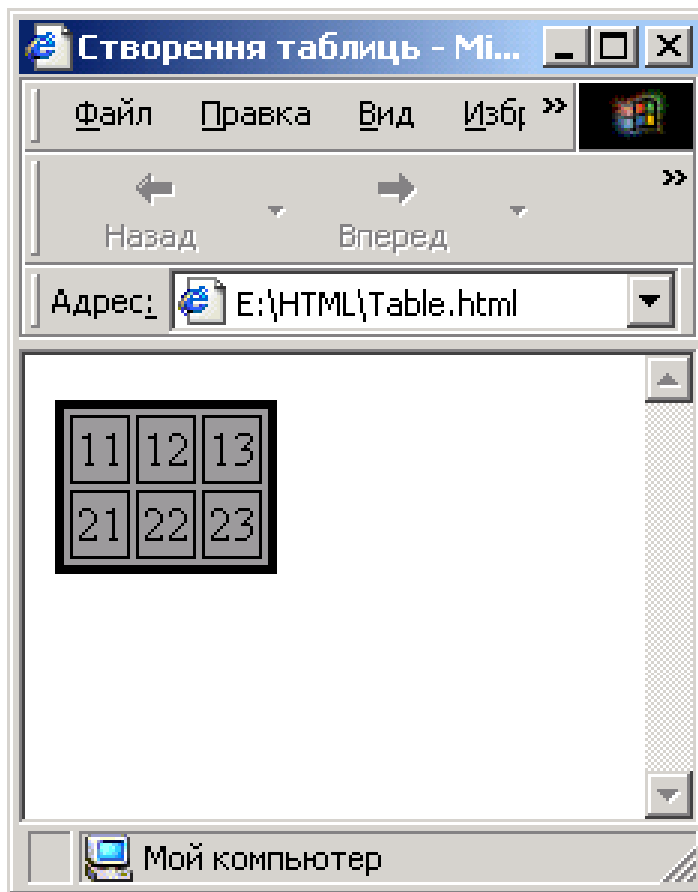



Рис. 2.43 Відображення таблиці із сірим кольором фону

6. Записати HTML-код для таблиці із білим та сірим кольором фону для сусідніх комірок та переглянемо її у браузері (рис. 2.44):

```

...
<table border="3" bordercolor="#000000">
  <tr>
    <td bgcolor="#ffffff">11</td>
    <td bgcolor="#999999">12</td>
    <td bgcolor="#ffffff">13</td>
  </tr>
  <tr>
    <td bgcolor="#999999">21</td>
    <td bgcolor="#ffffff">22</td>
    <td bgcolor="#999999">23</td>
  </tr>
</table>
</body>

```

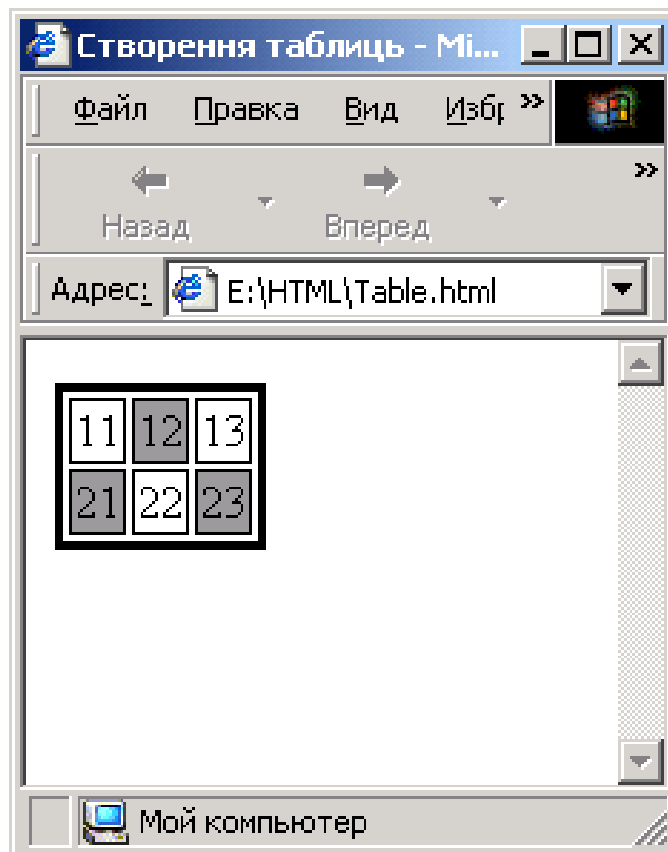


Рис. 2.44. Відображення таблиці із білим та сірим кольором фону для сусідніх комірок

7. Визначити висоту рядків та ширину стовпців. Створити таблицю із рядками висотою 40 та стовбцями шириною стовпців 60 пікселів. Ширина стовпців та висота рядків задається за допомогою параметрів `width` та `height` відповідно. Записати HTML-код для заданої таблиці та переглянути її у браузері (рис. 2.45):

```
...  
<table border="3" bordercolor="#000000">  
  <tr>  
    <td height = "40" width = "60" bgcolor = "#ffffff">11</td>  
    <td width = "60" bgcolor = "#999999">12</td>  
    <td width = "60" bgcolor = "#ffffff">13</td>  
  </tr>  
  <tr>  
    <td height = "40" width = "60" bgcolor = "#999999">21</td>  
    <td width = "60" bgcolor = "#ffffff">22</td>  
    <td width = "60" bgcolor = "#999999">23</td>  
  </tr>  
</table>  
</body>
```

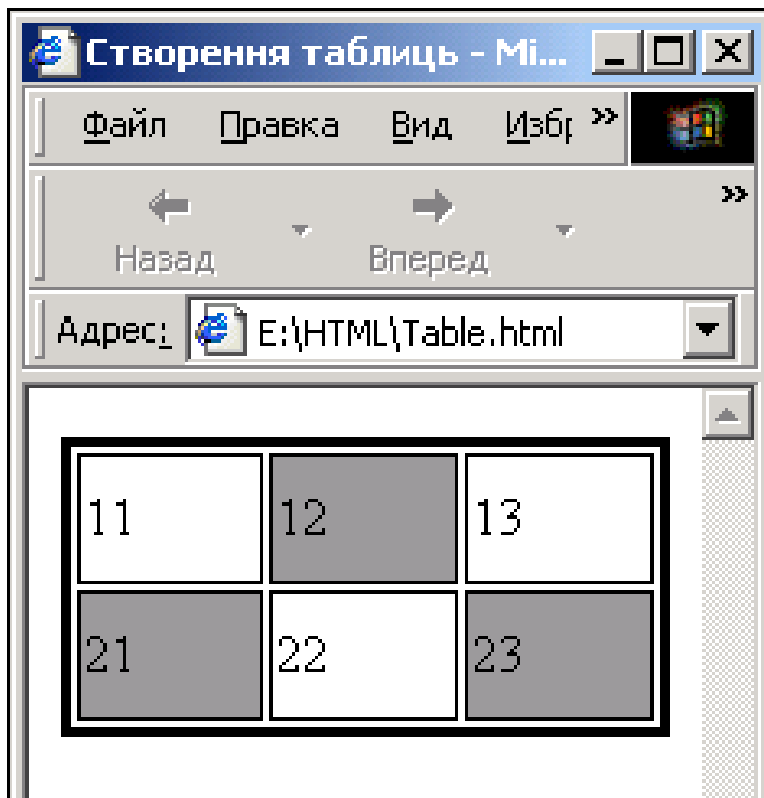


Рис. 2.45. Відображення таблиці із висотою рядків 40 та шириною стовпців 60 пікселів

8. Визначити висоту та ширину для всієї таблиці, тоді всі комірки (стовпці) та рядки поділять даний їм простір порівну, якщо не задавати їм особисто цього просторе у відсотках чи пікселях. Записати HTML-код для такої таблиці та переглянути її у браузері (рис. 2.46):

```

...
<table border="3" bordercolor="#000000" bgcolor = "#999999" height =
"200" width = "200">
  <tr>
    <td>11</td><td>12</td><td>13</td>
  </tr><tr>
    <td>21</td><td>22</td><td>23</td>
  </tr>
</table>
</body>

```

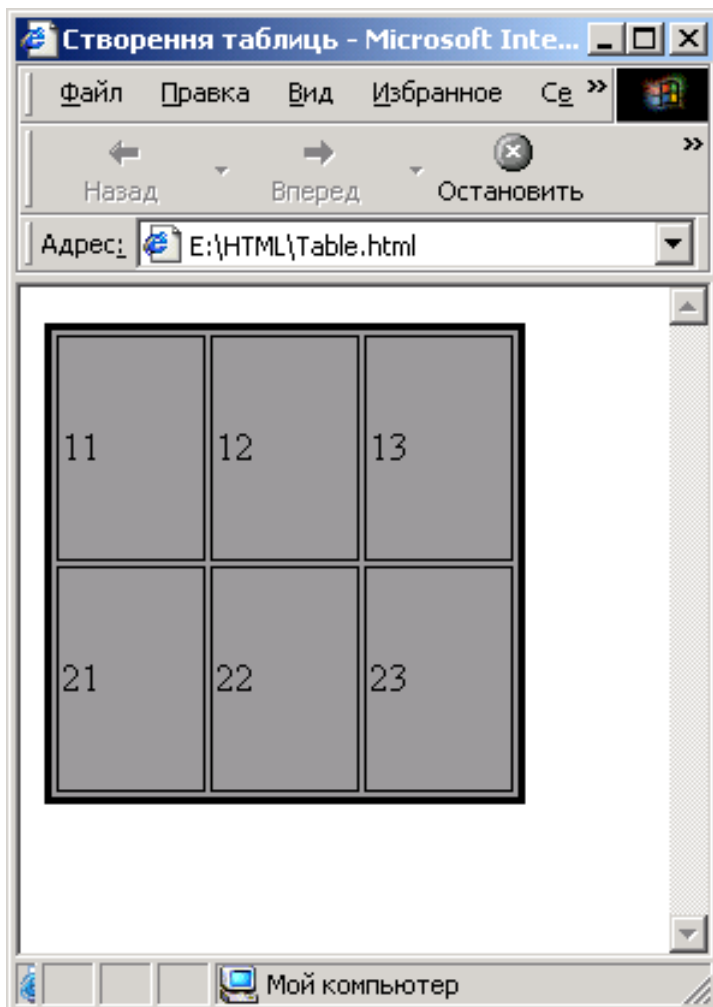


Рис. 2.46. Відображення таблиці із висотою 200 та шириною 200 пікселів

9. Для таблиці, представленої на рис. 2.46, визначити висоту рядків та ширину стовбців у відсотках до висоти та ширини всієї таблиці. Записати необхідний HTML-код та переглянути таблицю у браузері (рис. 2.47):

```

...
<table border="3" bordercolor="#000000" height = "200" width = "200" >
  <tr>
    <td height = "20%" width = "20%" bgcolor = "#ffffff">11</td>
    <td width = "30%" bgcolor = "#999999">12</td>
    <td width = "50%" bgcolor = "#ffffff">13</td>
  </tr><tr>
    <td height="80%" width="20%" bgcolor = "#999999">21 </td>
    <td width = "30%" bgcolor = "#ffffff">22</td>
    <td width = "50%" bgcolor = "#999999">23</td>
  </tr>
</table>
</body>

```

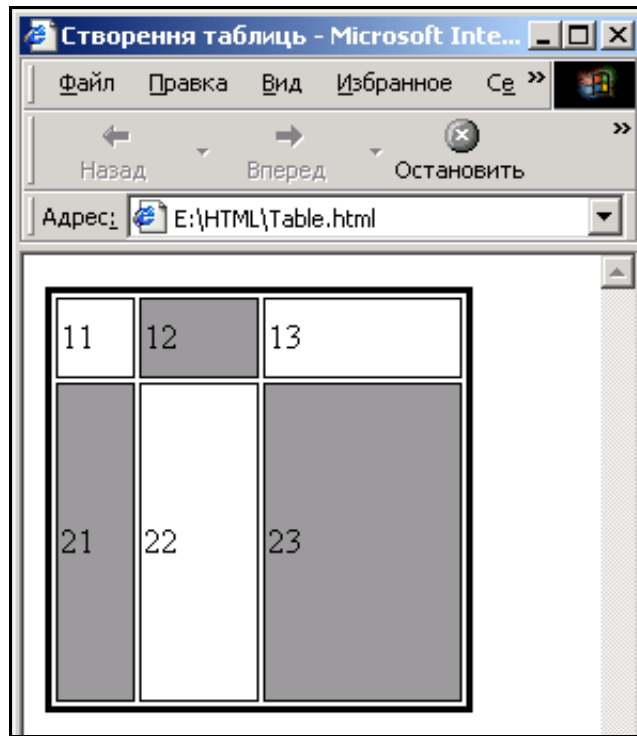


Рис. 2.47. Відображення таблиці із висотою та шириною стовпців, визначених у відсотках до висоти та ширини всієї таблиці

10.Зробити вертикальне та горизонтальне вирівнювання змісту таблиці. Таблиця, яку необхідно отримати зображена на рис. 2.48.

11	12	13
21	22	23

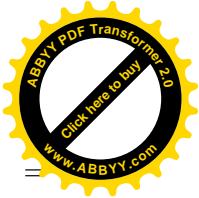
Рис. 2.48. Таблиця, яку необхідно відобразити у вікні браузера

Реалізувати це за допомогою параметру `valign="middle"` (top, bottom) для вертикального вирівнювання та параметру `align="center"` (left, right) для горизонтального вирівнювання. Записати HTML-код для заданої таблиці та переглянути її у браузері (рис. 2.49)

```

...
<table border="3" bordercolor="#000000" >
  <tr>
    <td height="40" width="60" valign="top" align="left" bgcolor =
      "#ffffff">11</td>
    <td width="60" valign="middle" align="center" bgcolor =
      "#999999">12</td>

```



```

<td width="60" valign="bottom" align="right" bgcolor =
"#ffffff">13</td>
</tr>
<tr>
<td height="40" width="60" valign="top" align="left" bgcolor
="#999999">21</td>
<td width="60" valign="middle" align="center" bgcolor =
"#ffffff">22</td>
<td width="60" valign="bottom" align="right" bgcolor =
"#999999">23</td>
</tr>
</table>
</body>

```

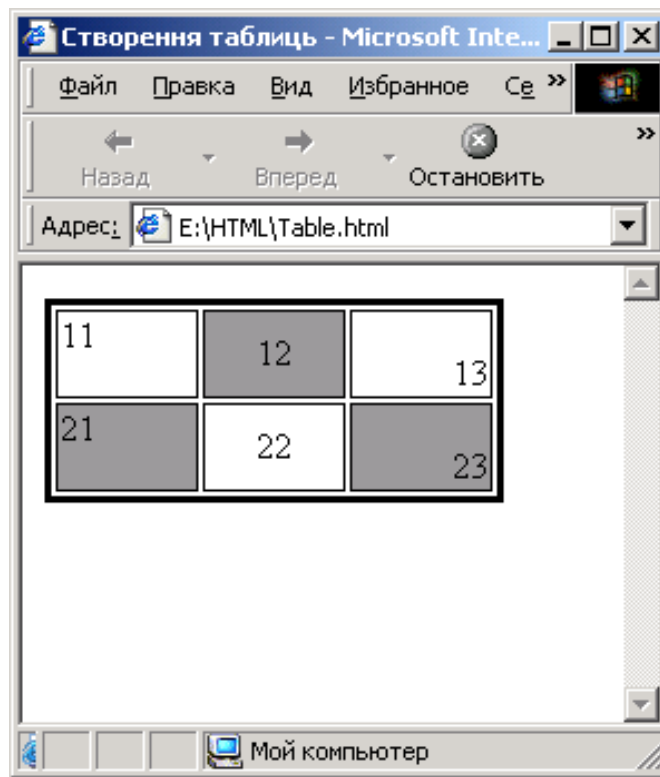


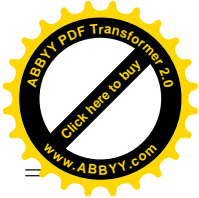
Рис. 2.49. Відображення таблиці з виконаним вирівнюванням змісту її комірок у вікні браузера

11. Об'єднати у першому рядку таблиці (рис.10) перші дві комірки та задати для них сірий колір фону. Для цього використаємо параметр colspan, який визначає кількість стовпців в об'єднаній комірці. Записати необхідний HTML-код:

```

...
<table border="3" bordercolor="#000000" >
<tr>
<td height="40" colspan="2" valign="top" align="left" bgcolor
= "#999999">11</td>

```



```
<td width="60" valign="bottom" align="right" bgcolor =  
"#ffffff">13</td>  
</tr>  
<tr>  
<td height="40" width="60" valign="top" align="left" bgcolor  
="#999999">21</td>  
<td width="60" valign="middle" align="center" bgcolor =  
"#ffffff">22</td>  
<td width="60" valign="bottom" align="right" bgcolor =  
"#999999">23</td>  
</tr>  
</table>  
</body>
```

Відповідна таблиця представлена на рис. 2.50.

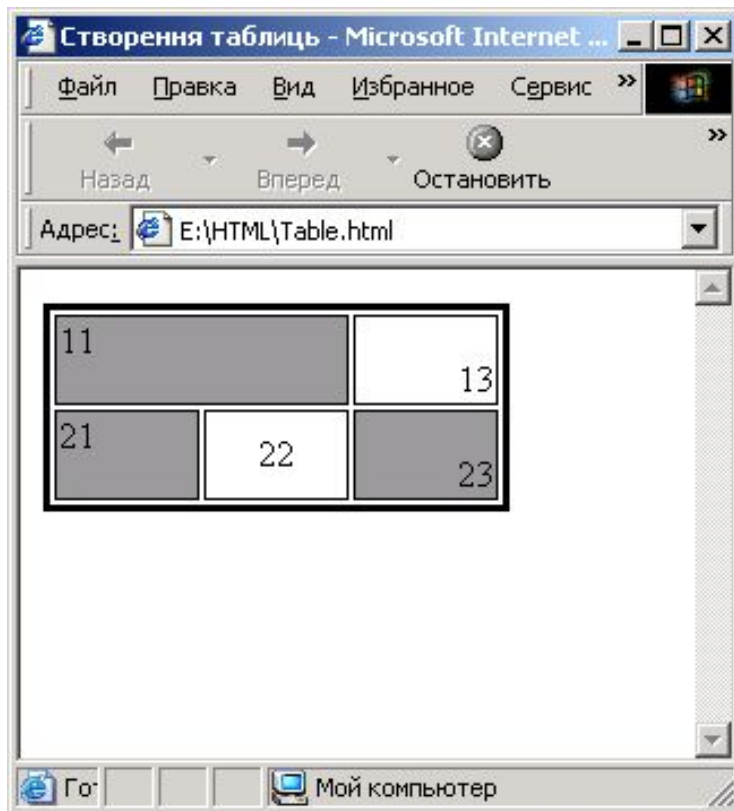


Рис. 2.50. Відображення таблиці з об'єднаними у першому рядку першими двома комірками, для яких заданий сірий колір фону.

12. Використання параметрів cellpadding та cellspacing.

– Встановити відстань між комірками таблиці, представленої на рис. 2.50, рівною 0 пікселів (це білий простір між комірками таблиці). Реалізувати це за допомогою параметру cellpadding. Записати необхідний HTML-код:



```
...  
<table border="3" bordercolor="#000000" cellspacing="0" >  
  <tr>  
    <td height="40" colspan="2" valign="top" align="left" bgcolor  
    = "#999999">11</td>  
    <td width="60" valign="bottom" align="right" bgcolor =  
    "#ffffff">13</td>  
  </tr>  
  <tr>  
    <td height="40" width="60" valign="top" align="left" bgcolor  
    = "#999999">21</td>  
    <td width="60" valign="middle" align="center" bgcolor =  
    "#ffffff">22</td>  
    <td width="60" valign="bottom" align="right" bgcolor =  
    "#999999">23</td>  
  </tr>  
</table>  
</body>
```

Відповідна таблиця показана на рис. 2.51.

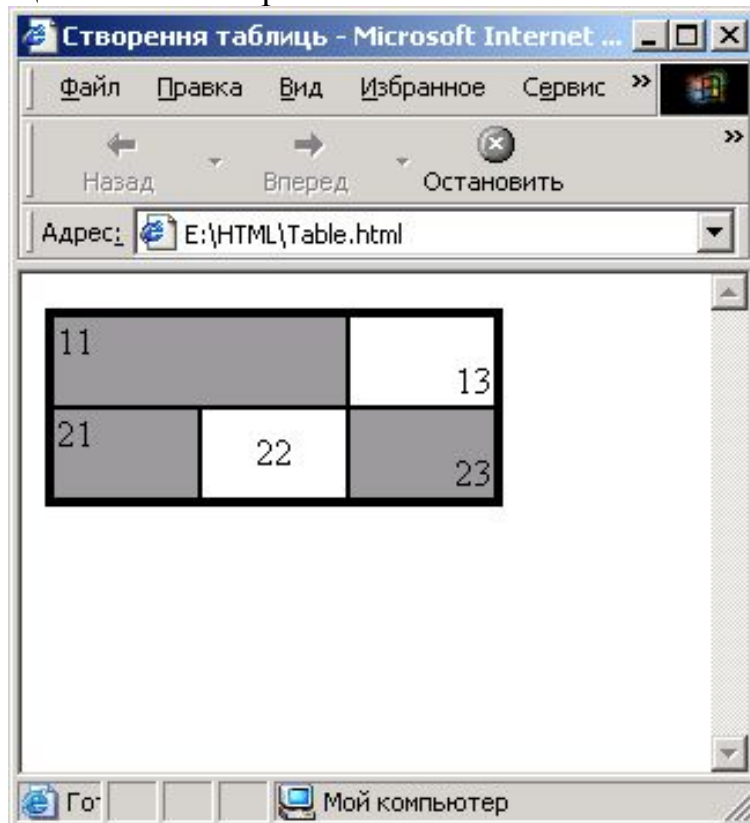


Рис. 2.51. Відображення таблиці, в якій відстань між комірками дорівнює 0 пікселів

– Змінити значення параметру cellspacing з 0 на 7. Нова таблиця показана на рис. 2.52.

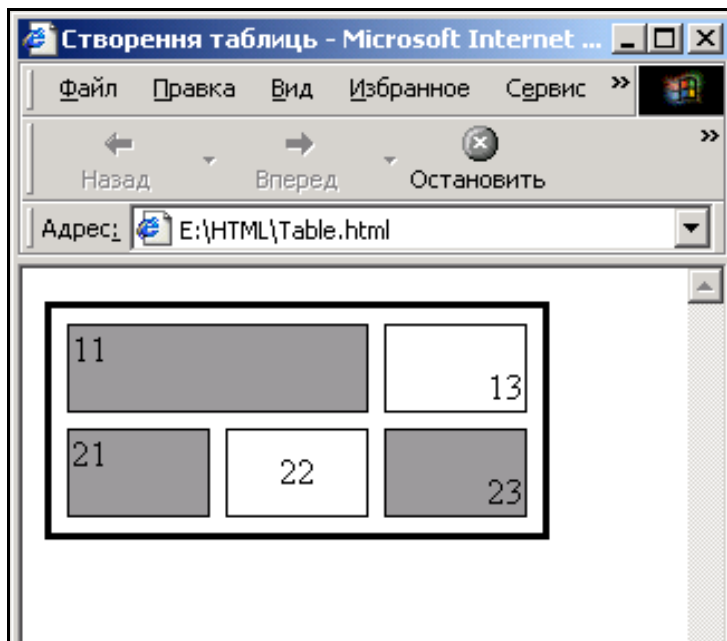


Рис. 2.52. Відображення таблиці, в якій відстань між комірками дорівнює 7 пікселів

– Розглянути наслідки використання параметру `cellpadding`, що визначає відстань між границями комірок і даними, що знаходяться в цих комірках. Для наглядності результатів спочатку за допомогою параметру `valing` вирівняти зміст комірок верхнього ряду доверху, а нижнього-донизу:

...

```

<table border="3" bordercolor="#000000" cellpadding="0" >
  <tr>
    <td height="40" colspan="2" valign="top" align="left"
      bgcolor = "#999999">11</td>
    <td width="60" valign="top" align="right" bgcolor =
      "#ffffff">13</td>
  </tr>
  <tr>
    <td height="40" width="60" valign="bottom" align="left"
      bgcolor = "#999999">21</td>
    <td width="60" valign="bottom" align="center" bgcolor =
      "#ffffff">22</td>
    <td width="60" valign="bottom" align="right" bgcolor =
      "#999999">23</td>
  </tr>
</table>
</body>

```

Відповідна таблиця показана на рис. 2.53.

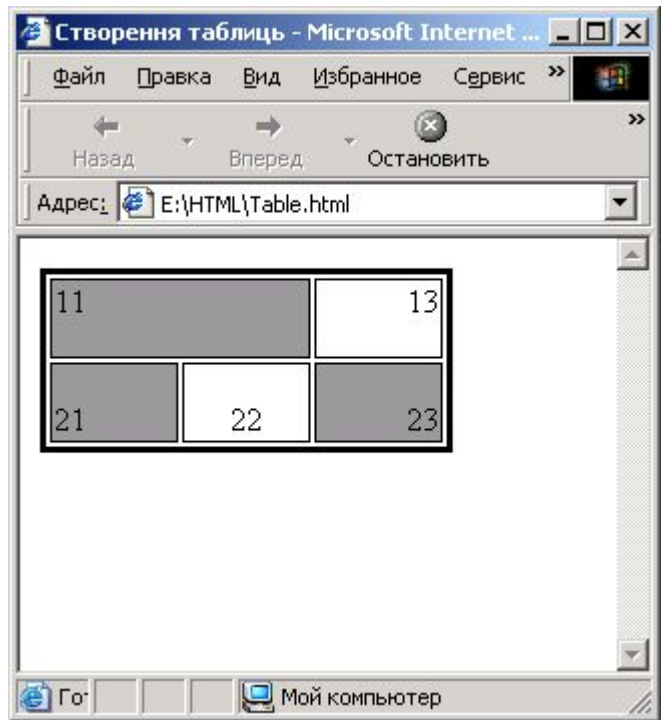


Рис. 2.53. Відображення таблиці, для якої параметр cellpadding дорівнює 0 пікселів.

Змінити значення параметру cellpadding з 0 на 7. Нова таблиця показана на рис. 2.54.

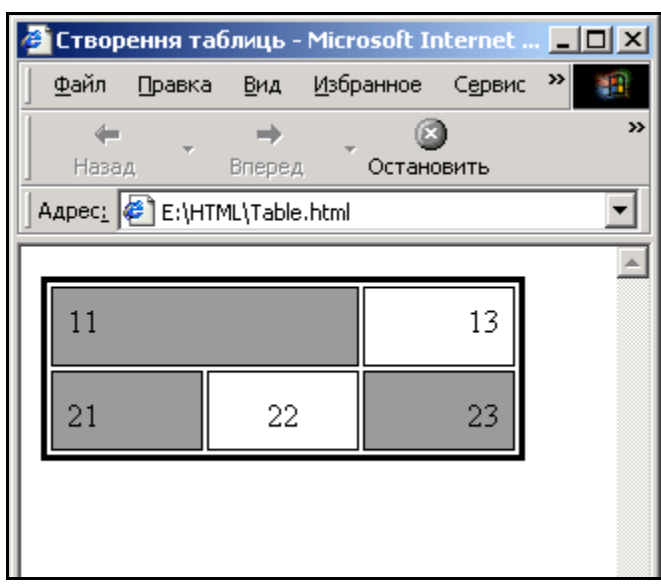


Рис. 2.54. Відображення таблиці, для якої параметр cellpadding дорівнює 7 пікселів.

13. Визначити фоновий рисунок таблиці (рис. 2.55). Реалізувати це за допомогою параметру background. HTML-код при цьому буде такий:

```
...  
<table height = "200" width = "200" background="Zamok.jpg">  
  <tr>
```

```

        <td></td><td>Замок</td><td></td>
    </tr>
    <tr>
        <td></td><td></td><td></td>
    </tr>
</table>
</body>

```

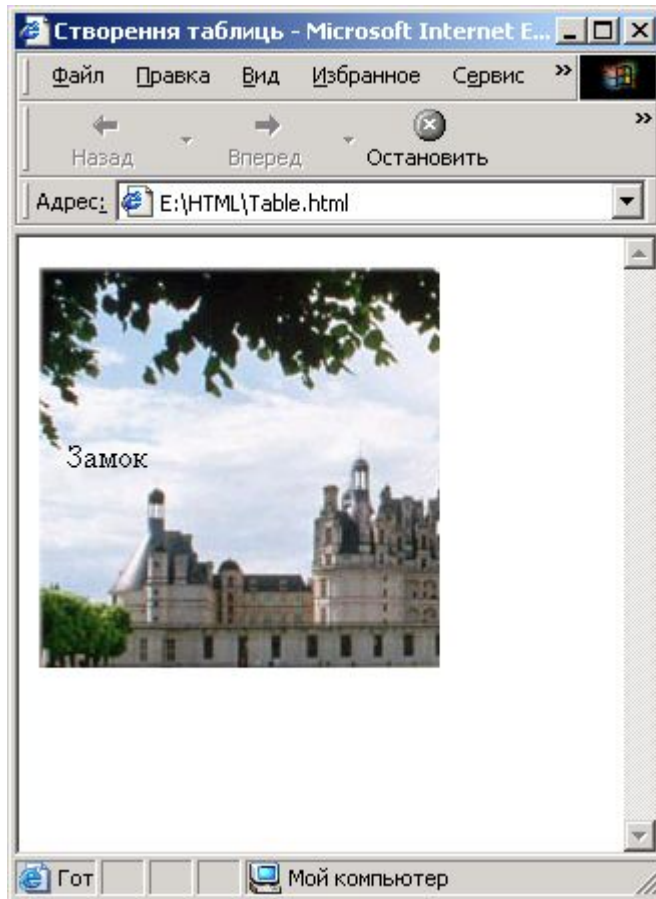


Рис. 2.55. Відображення таблиці, для якої визначено параметр background.

14. Вкладені таблиці. Створити таблицю, яка складається з одного ряду та трьох комірок. В третю комірку додати ще одну таблицю, яка складається з чотирьох рядків та двох стовпців. HTML- код для такої таблиці такий:

```

...
<table align="center">
  <tr height="200">
    <td width="200" valign="top" background="Zamok.jpg" align="center">
      <h1><br><br>3BIT</h1></td>
    <td width="50" background="Tochka.jpg"></td>
    <td width="300" valign="top" background="Yellow.jpg" align="center">
      <h3>Таблиця, яка показує успішність групи ОА-Д-2:</h3>
      <table cellpadding="3" border="3" bordercolor="black"
        background="Yellow.jpg">
        <tr height="40">

```

```

<td width="200" align="center"> п'ятірок </td>
<td width="100" align="center">10%</td>
</tr>
<tr height="40">
<td width="200" align="center">четвірок </td>
<td width="100" align="center">70% </td>
</tr>
<tr height="40">
<td width="200" align="center">тріюк </td>
<td width="100" align="center">20% </td>
</tr>
<tr height="40">
<td width="200" align="center">двійок </td>
<td width="100" align="center">не має </td>
</tr>
</table>
<br> <h3>Викладач</h3>
</td>
</tr>
</table>
</body>

```

Переглянути цю таблицю у браузері (рис. 2.56)

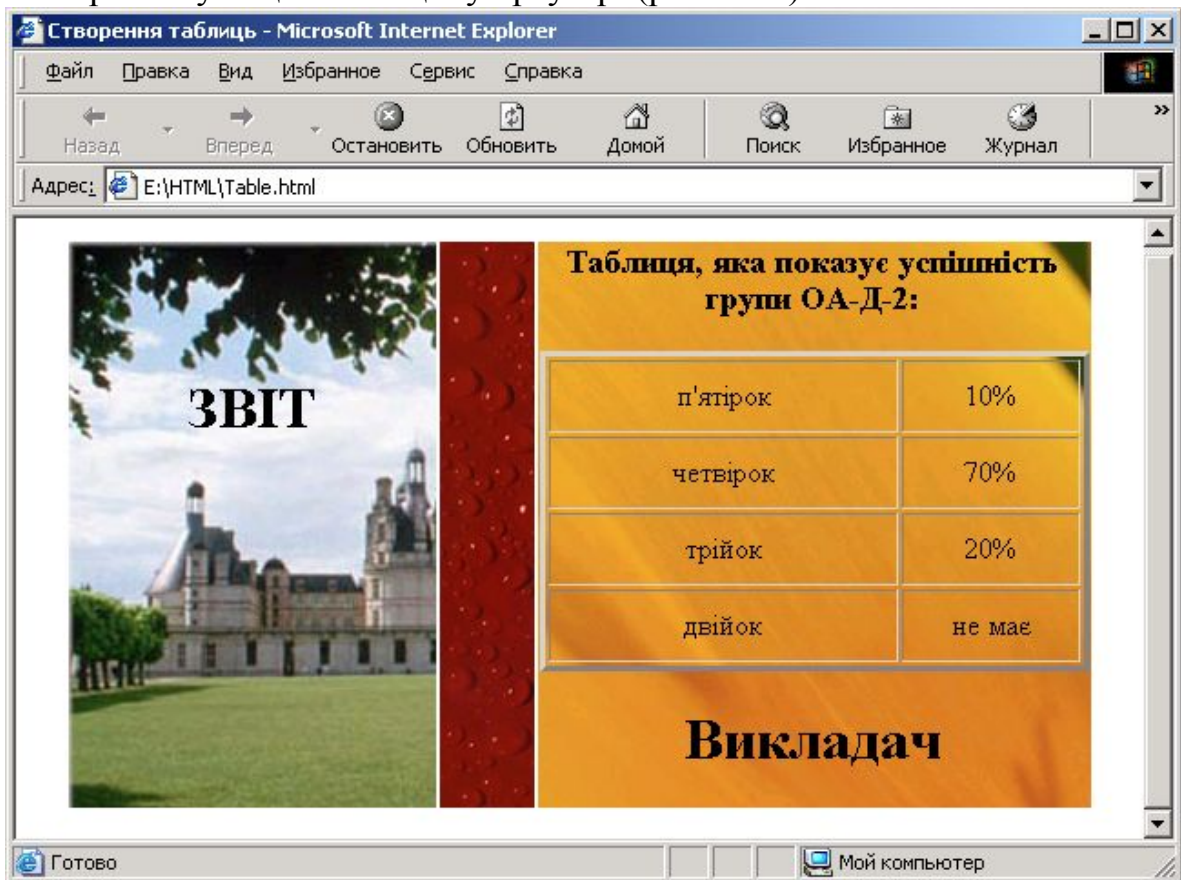


Рис. 2.56 . Відображення таблиці, яка складається з одного ряду та трьох комірок та містить вкладену таблицю в третій комірці.



Варіанти завдань до лабораторної роботи № 6

Варіант № 1

Позначення механізмів	Кількість комплектів	Добуток за рік		Різниця
		Планова	Фактична	
M138	5	3,75	2,5	-6,25
M231	2	4,91	5,1	0,38
K145	7	7,31	7,11	-1,4
K011	10	5,25	6	7,5

Варіант № 2

Вироби	Вартість 1 виробу		Кількість в 1999 році	Загальна вартість в 1999 році
	В 1998 році	В 1999 році		
Виріб А	35	32	150	4800
Виріб Б	50	46	200	9200
Виріб В	70	60	120	7200
Виріб Г	27	25	300	7500

Варіант № 3

Позначення механізмів	Продуктивність добова		Кількість суток	Економія підсумкова
	Планова	Фактична		
1МА	350	360	50	500
1МБ	500	530	20	600
2МТ	470	480	10	100
1ТМ	125	150	30	750

Варіант № 4

Назва товару	Кількість			Залишилось
	Залишок	Надійшло	Продано	
Зошит	100	500	550	50
Лінійка	50	20	15	55
Ручка	150	200	120	230
Блокнот	20	100	70	50

Варіант № 5

Позначення механізмів	Загрузка добова		Кількість діб	Процент загрузки
	Планова	Фактична		
МА10	11	10	7	91%
МБ20	20	20	10	100%
МТ01	25	20	5	80%
ТМ05	18	16	2	89%



Варіант № 6

Прізвище, ім'я, по батькові	Шифр	Начислено		Усього
		Оклад	Надбавка	
Іванов І.І.	3150	350	100	450
Сомов С.С.	1220	500	200	700
Титов Т.Т.	2110	470	180	650
Котов К.К.	1530	125	50	175

Варіант № 7

Номер рахунку	Вид платежа		Адреса	Сума платежу
	Квартплата	За світло		
112233	110	22,50	Донецьк-50	132,50
123456	70	11	Донецьк-10	81,00
223344	35	10	Донецьк-111	45,00
554433	85	15,50	Донецьк-2	100,50

Варіант № 8

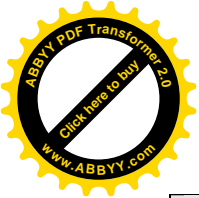
Назва станка	Шифр	Зайнятість обладнання		Резерв
		В наявності	Використовується	
Карусельний	403150	75	50	25
Токарний	321220	50	40	10
Фрезерний.	702110	47	38	9
Шліфувальний	251530	25	20	5

Варіант № 9

Прізвище студента	Предмет			Номер студбілета	Підсумковий бал
	Математика	Фізика	Англійський		
Петров П.П.	5	4	5	981122	14
Ванін В.В.	4	4	4	981133	12
Котов К.К.	3	4	5	981144	12
Лосев Л.Л.	3	3	5	981135	11

Варіант № 10

Вид матеріалу	Ціна за 1 тону	Витрати матеріалу			
		Станки		Машини	
		Тонн	Грн.	Тонн	Грн.
Чавун	50	0,400	20,0	0,800	40,00
Сталь1	110	0,038	4,18	0,070	7,70
Прокат	90	0,850	76,50	2,500	225,00
Сталь2	100	0,050	5,00	0,100	10,00



Варіант № 11

Шифр виробу	Випуск продукції по заводам			Усього
	Завод1	Завод2	Завод3	
5135	300	550	250	1100
2225	750	350	800	1900
1001	810	220	600	1630
7500	270	225	200	695

Варіант № 12

Заводи	Основні фонди			Підсумок
	Було	Введено	Вибуло	
Завод1	12,3	1,2	0,6	12,9
Завод2	15,7	2,1	0,9	16,9
Завод3	21,1	3,5	1,7	22,9

Варіант № 13

Основні фонди	Показники		Приріст
	На початок року	На кінець року	
Будови виробничі	1137	1415	278
Будови жилі	651	685	34
Машини й обладнання	950	1120	170
Господарський інвентар	24	30	6

Варіант № 14

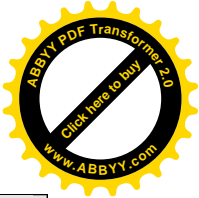
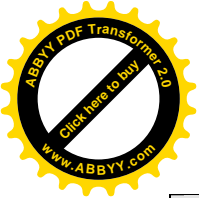
Номер креслення	Номер операції	Деталей		Усього
		Було	Випущено	
333444555	50	185	50	235
223344556	20	250	40	290
111222333	10	145	70	215

Варіант № 15

Назва деталей	Випущено в 1997 році	Затрати на випуск 1 дет.		Економія підсумкова
		В 1996 році	В 1997 році	
Поршневі кільця	1200	12,0	6,0	7200
Поршневі пальці	2200	2,8	1,9	1980
Насоси	50	1,5	0,7	40
Клапани	3700	14,8	10,4	16280

Варіант № 16

Назва шахти	Середньомісячна продуктивність		Процент виконання плану
	План	Випуск	
Кіровська	55,6	43,8	78,78%
Трудівська	60,8	71,5	117,60%
Заперевальна	57,4	58,9	102,61%
Лідівка	41,8	40,5	96,89%



Варіант № 17

Назва шахти	Показники		Продуктивність праці
	Добуток вугілля	Чисельність роб.	
ім. Горького	2055	1057	1,94
ім. Засядько	3860,8	1750	2,21
ім. Абакумова	1950	1250	1,56
ім. Скочинського	2180	1125	1,94

Варіант № 18

Завод	Відомості			Витрати на Зарплату
	Число робочих	Середня зарплата	Місяць	
Завод А	3500	290	січень	1015000
Завод Б	2000	210	січень	420000
Завод В	2300	180	січень	414000
Завод Г	4200	250	січень	1050000

Варіант № 19

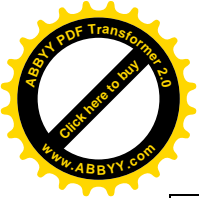
Назва шахти	Середньодобовий добуток		Перевиконано
	Планова	Фактична	
ім. Горького	3635	3050	-585
ім. Засядько	5270	5920	650
Кіровська	3200	3150	-50
Трудівська	5835	6070	235

Варіант № 20

Назва товару	Кількість	Ціна за 1 шт.		Прибуток підсумковий
		Оптом	В роздріб	
Зошит	200	0,10	0,20	20
Блокнот	100	2,50	3,20	70
Ручка	150	1,10	1,50	60
Олівець	120	0,25	0,40	18

Варіант № 21

Позначення механізмів	Кількість діб	Нагрузка добова		Перегрузка підсумкова
		Допустима	Фактична	
1МТ	10	1000	1050	500
2МХ	15	500	510	150
2ТМ	5	1200	1220	100
1ХМ	20	750	790	800



Варіант № 22

Прізвище студента	Кількість оцінок			Усього "4" и "5"
	"5"	"4"	"3"	
Попов П.П.	25	10	7	35
Іванов І.І.	6	17	8	23
Зотов В.В.	28	15	0	43
Сомов А.А.	3	12	30	15

Варіант № 23

Назва книги	Клас	По ведомости		Сума
		Ціна 1 шт.	Кількість	
Історія	7	6,40	20	128,00
Фізика	9	7,30	15	109,50
Англійський	8	7,10	25	177,50
Математика	5	5,90	40	236,00

Варіант № 24

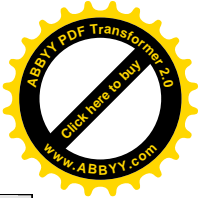
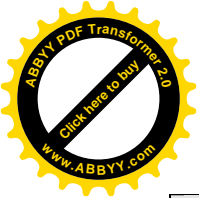
Прізвище, ім'я, по батькові	Кількість виходів	Начислено		Усього
		За 1 вихід	Надбавка	
Петров М.М.	20	5,50	100	210,00
Сомов С.С.	15	7,50	120	232,50
Тітов Т.Т.	24	9,20	200	420,80
Ванін К.К.	18	6,40	110	225,20

Варіант № 25

Прізвище, ім'я, по батькові	Номер рахунка	На рахунку		Прибуток
		На початок року	На кінець року	
Лось Л.Л.	11223344	1200	1300	100
Васін В.В.	22334455	1850	2050	200
Волков С.С.	12345678	920	1070	150
Петров А.А.	11335577	2350	2700	350

Варіант № 26

Назва книги	Ціна 1 екземпляра	Кількість		Виручка
		Було	Залишилось	
Хімія	5,20	50	20	156,00
Математика	8,10	70	15	445,50
Англійський	7,30	45	25	146,00
Історія	4,90	60	40	98,00



Варіант № 27

Назва книги	Ціна 1 екземпляра	Кількість		Залишилось на суму
		Було	Продали	
Казки	4,70	100	35	305,50
Вірші	2,90	70	20	145,00
П'єси	5,40	40	10	162,00
Історії	4,30	55	25	129,00

Варіант № 28

Назва деталі	Кількість	Оплатити		Сума
		За 1 шт.	Премія	
Шестерня	120	1,70	20	224
Картер	50	2,30	10	125
Вал	70	5,10	15	372
Підшипник	100	7,60	40	800

Варіант № 29

Назви ліків	Ціна 1 упаковки	Кількість упаковок		Виручка
		Було	Залишилось	
Анальгін	0,80	50	25	20,00
Цефекон	7,20	40	20	144,00
Нафтізін	2,10	20	5	31,50
Аспірин	3,40	70	35	119,00

Варіант № 30

Назви ліків	Кількість упаковок			Усього
	Було	Отримано	Витрачено	
Аспірин	20	15	30	5
Вітаміни	50	40	70	20
Валідол	35	40	20	55
Йод	15	40	45	10

Контрольні питання:

1. Навіщо використовуються таблиці?
2. Як визначити кількість рядків у таблиці?
3. Як визначити кількість колонок в таблиці?
4. Як визначити горизонтальне вирівнювання змісту комірок таблиці?
5. Чи можна визначити вертикальне вирівнювання змісту рядків таблиці?
6. Чи можна визначити ширину окремої комірки таблиці?
7. Як визначити ширину таблиці?
8. Чи можна визначити висоту таблиці?
9. Як встановити відстань між комірками таблиці?
10. Як встановити товщину границь таблиці?



2.7 Лабораторна робота №7 "Створення фреймів "

Мета роботи: навчитись створювати фрейми.

Завдання:

Визначити фрейм з структурою, вказаною викладачам.

Методичні вказівки.

1. Скопіювати в теку HTML виданий викладачем графічний файл Eifel.jpg
2. В теці HTML створити текстовий документ з назвою Frame.html.
3. У файлі Frame.html визначити чотири фрейми, з структурою показаною на рис. 2.57

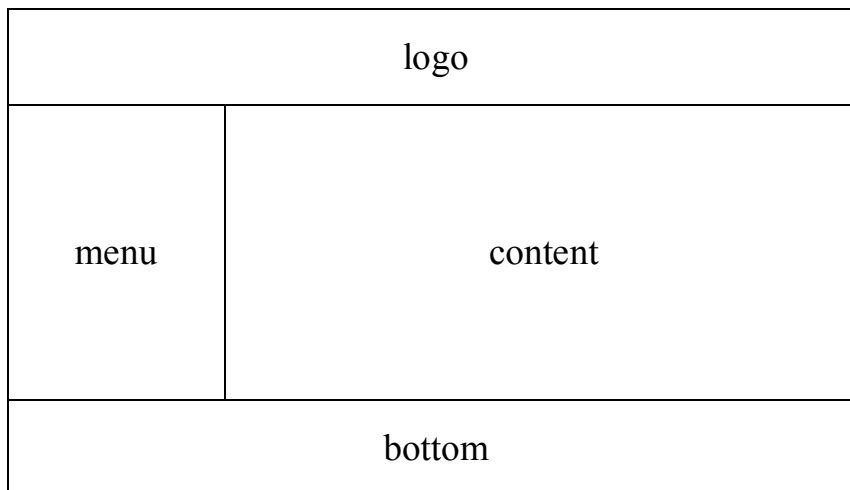


Рис. 2.57. Розміщення фреймів у вікні браузера

Ім'я першого фрейму logo, він займає всю ширину сторінки, та в ньому завантажений файл logo.html. Далі ідуть два центральні фрейми: menu, в ньому завантажений файл menu.html та content, в ньому завантажений файл content.html. Фрейм menu займає 25% у ширину, а фрейм content – всій простір, що залишився. Останній фрейм bottom займає нижню частину екрану, в ньому завантажений файл bottom.html. В файл Frame.html записати HTML-код для визначення фреймової структури:

```
<html>
<head>
  <title>Створення фреймів</title>
</head>
<frameset rows="25%,50%,25%">
  <frame src="logo.html">
    <frameset cols="25%,*">
      <frame src="menu.html">
      <frame src="content.html">
```



```
        </frameset>
        <frame src="bottom.html">
</frameset>
</html>
```

4. В теці HTML створити текстові документи з назвами logo.html, menu.html, content.html, bottom.html, та записати в них такий HTML-код:

– HTML-код документу logo.html:

```
<html>
<head>
  <title>logo</title>
</head>
<body >
  
  <h2 align="center">ТУРИСТИЧНА ФІРМА "АРС"</h2>
</body>
</html>
```

– HTML-код документу menu.html:

```
<html>
<head>
  <title>menu</title>
</head>
<body>
  Новини<br><br>
  Ціни<br><br>
  Погода
</body>
</html>
```

– HTML-код документу content.html:

```
<html>
<head>
  <title>content</title>
</head>
<body>
  <center>Зміст документу</center>
</body>
</html>
```

– HTML-код документу bottom.html:

```
<html>
<head>
  <title>bottom</title>
</head>
<body>
```



```
<center>Нижній колонтитул документа</center>
</body>
</html>
```

Відображення фреймів (файлу Frame.html) в браузері повинно відповідати рис. 2.58.

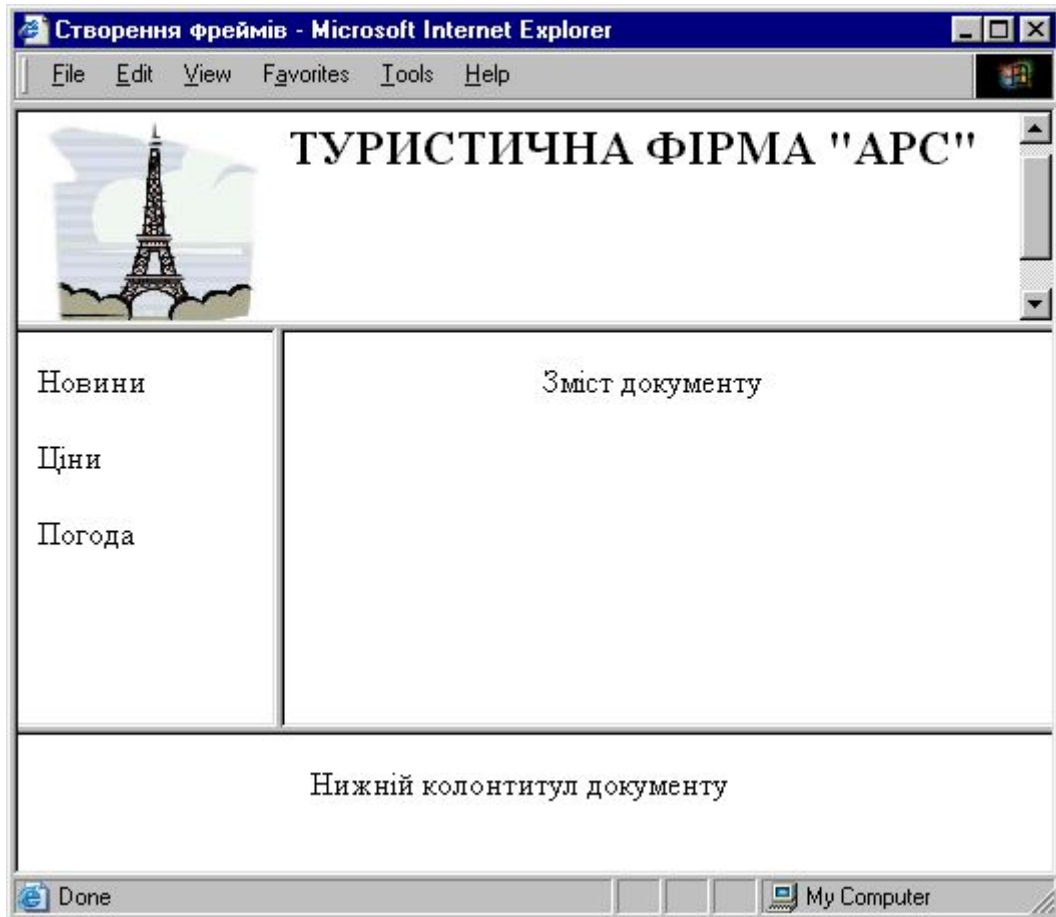


Рис. 2.58. Відображення HTML-документа з чотирма фреймами у вікні браузера

5. Зробити так, щоб смуга прокрутки у фреймі logo була відсутня. Реалізувати це за допомогою параметру *scrolling* тегу `<frame>`, значення якого в даному випадку буде дорівнювати "no". Відкорегувати HTML-код файлу Frame.html :

```
<frameset rows="25%,50%,25%">
<frame src="logo.html" scrolling="no">
  <frameset cols="25%,*">
    <frame src="menu.html">
    <frame src="content.html" >
  </frameset>
<frame src="bottom.html">
</frameset>
```

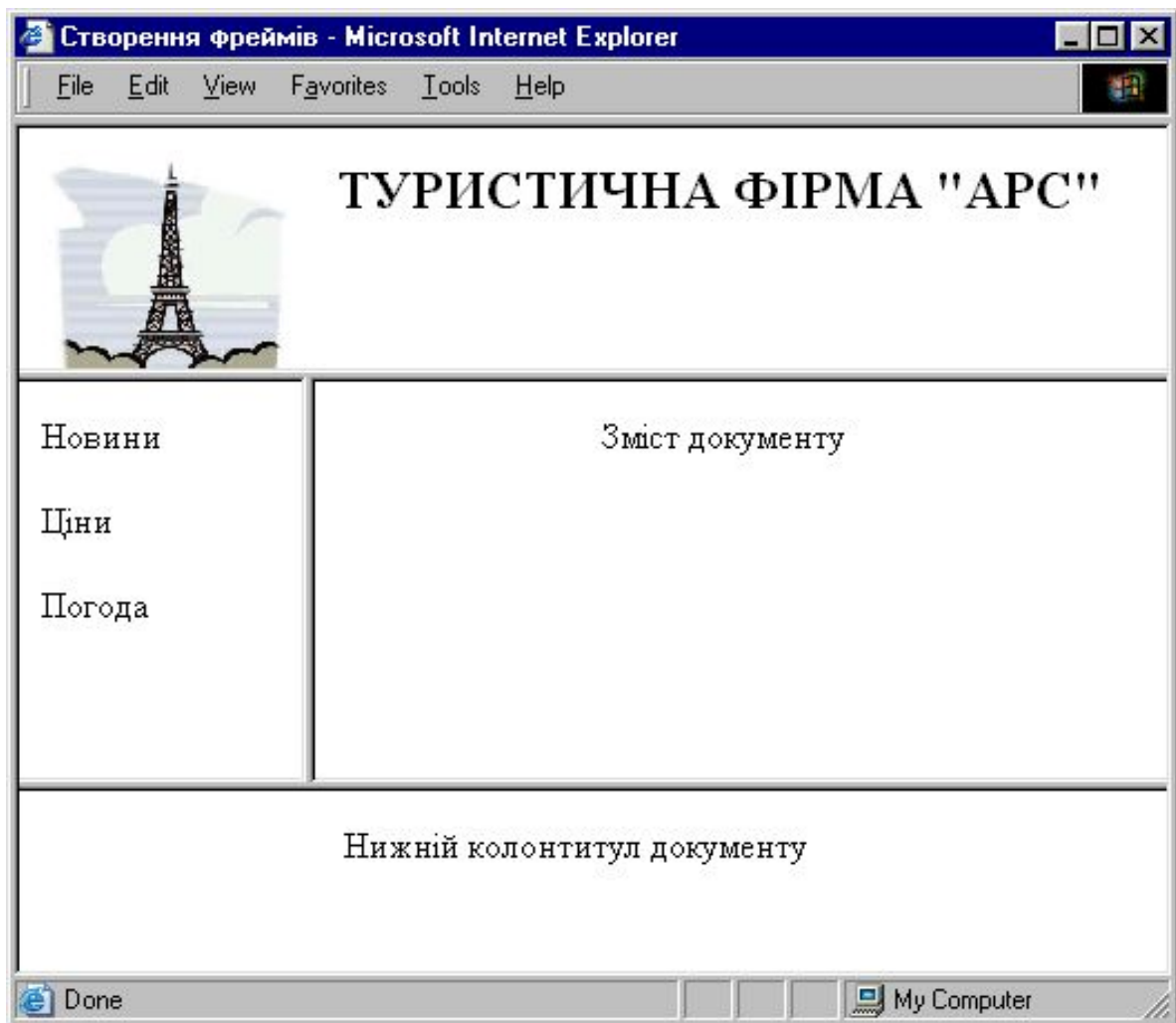


Рис. 2.59. Відображення у браузері чотирьох фреймів, в яких відсутні смуги прокрутки

6. Зробити так, щоб рамки між фреймами були відсутні. Реалізуємо це за допомогою параметру *frameborder* тегу `<frame>`, значення якого в цьому випадку дорівнює 0. Скорегувати HTML-код:

```
<frameset rows="25%,50%,25%">  
<frame src="logo.html" scrolling="no" frameborder=0>  
  <frameset cols="25%,*">  
    <frame src="menu.html" frameborder=0>  
    <frame src="content.html" frameborder=0>  
  </frameset>  
<frame src="bottom.html" frameborder=0>  
</frameset>
```

Переглянути цей документ у вікні браузера (рис. 2.60)

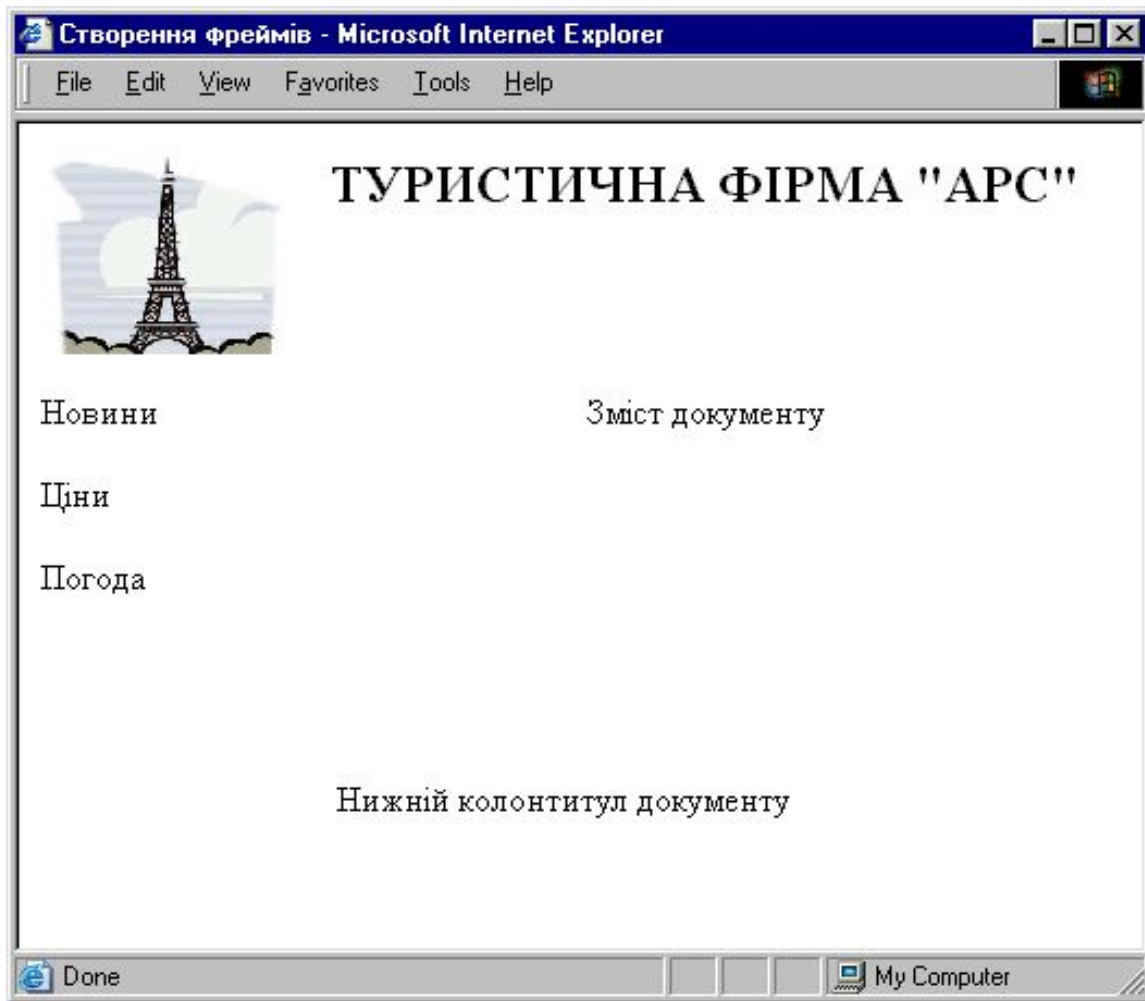


Рис. 2.60. Відображення у браузері чотирьох фреймів, для яких визначена відсутність межі між фреймами

7. Змінити значення параметру *frameborder* з 0 на 1, щоб визначити наявність межі між фреймами. А також визначити зелений колір межі між фреймами – це робити за допомогою параметру *bordercolor* тегу `<frame>`. Записати відповідний HTML-код:

```
<frameset rows="25%,50%,25%">
  <frame src="logo.html" scrolling="no" frameborder=1
bordercolor="green">
  <frameset cols="25%,*">
    <frame src="menu.html" frameborder=1 bordercolor="green">
    <frame src="content.html" frameborder=1 bordercolor="green">
  </frameset>
  <frame src="bottom.html" frameborder=1 bordercolor="green"> >
</frameset>
```

Переглянути документ Frame.html у браузері (рис. 2.61)

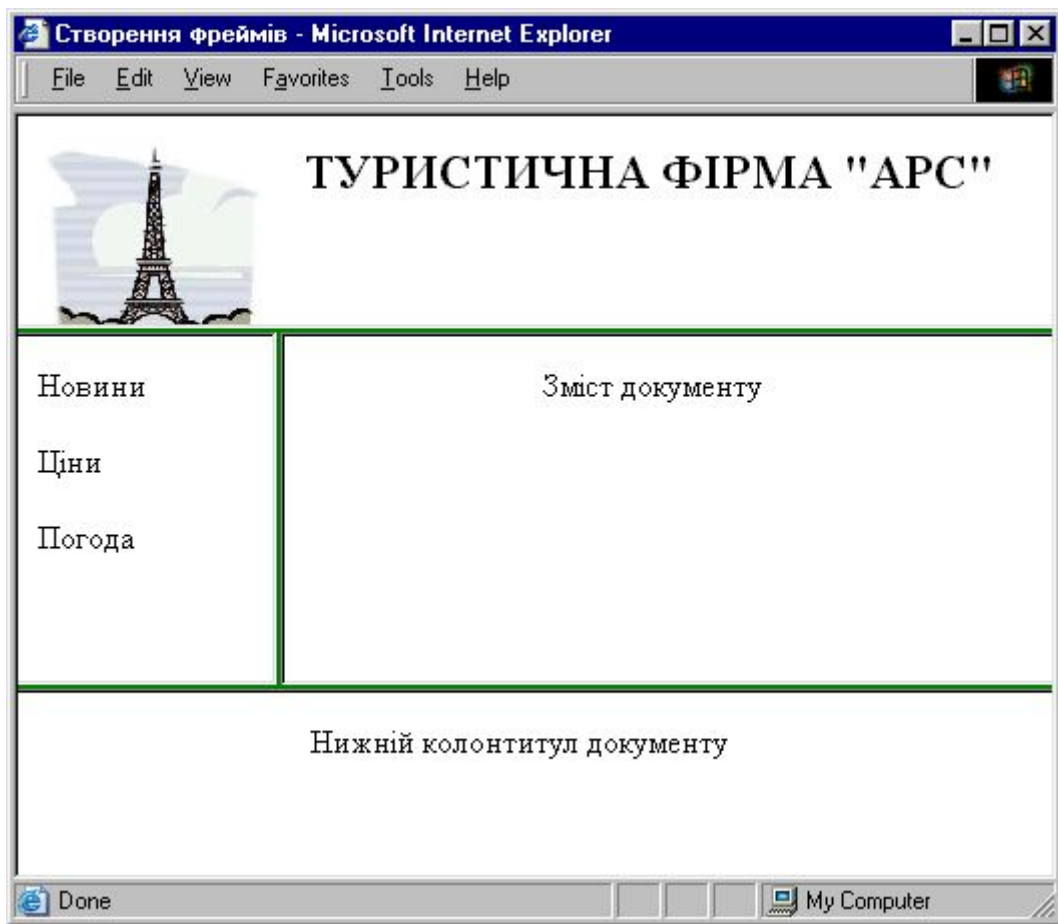


Рис. 2.61. Відображення у браузері чотирьох фреймів, для яких визначена наявність межі між фреймами та зелений колір меж

8. Визначити простір усередині кожного фрейму, тобто поля, в межах яких не може бути розміщена ніяка інформація. Реалізувати це за допомогою параметрів *marginheight* та *marginwidth*, значення яких буде дорівнювати 1 пікселю.

Записати HTML-код та переглянути документ у вікні браузера (рис. 2.62):

```
<frameset rows="25%,50%,25%">
<frame src="logo.html" scrolling="no" frameborder=1 bordercolor="green"
marginheight="1" marginwidth="1">
  <frameset cols="25%,*">
    <frame src="menu.html" scrolling="no" frameborder=1
bordercolor="green"marginheight="1" marginwidth="1">
    <frame src="content.html" frameborder=1 bordercolor="green"
marginheight="1" marginwidth="1">
  </frameset>
<frame src="bottom.html" frameborder=1
bordercolor="green"marginheight="1" marginwidth="1">
</frameset>
```

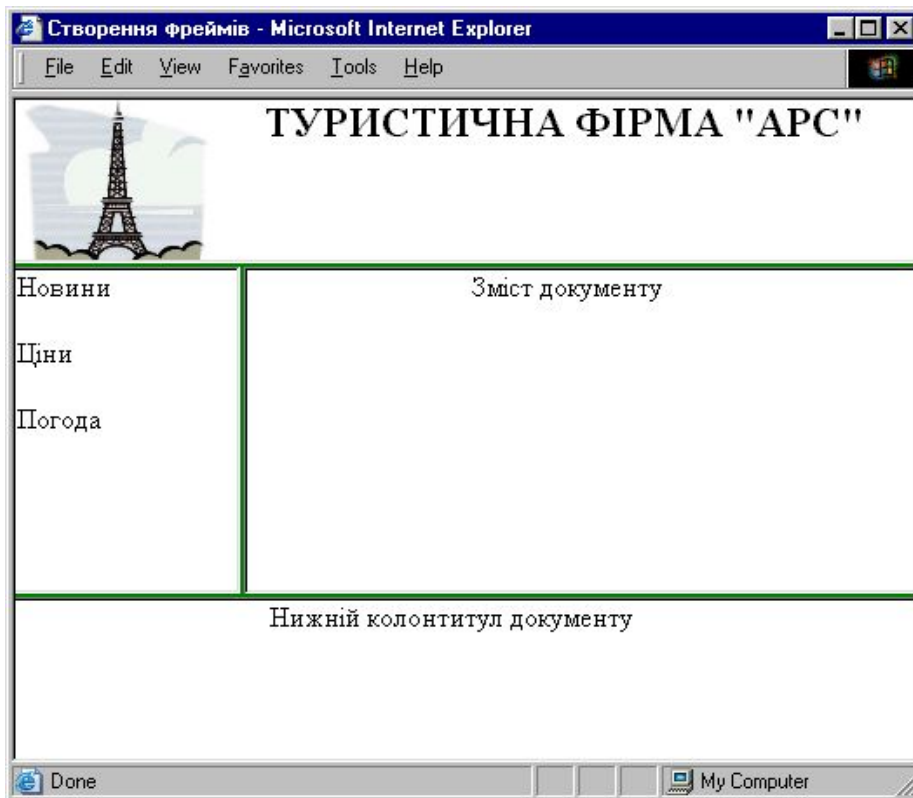



Рис. 2.62. Відображення у браузері чотирьох фреймів, для яких визначено параметри *marginheight* та *marginwidth*, значення яких дорівнює 1 пікселю

Змінити значення параметрів *marginheight* та *marginwidth* з 1 на 20 пікселів для фреймів logo та menu. Переглянути новий документ на рис. 2.63.

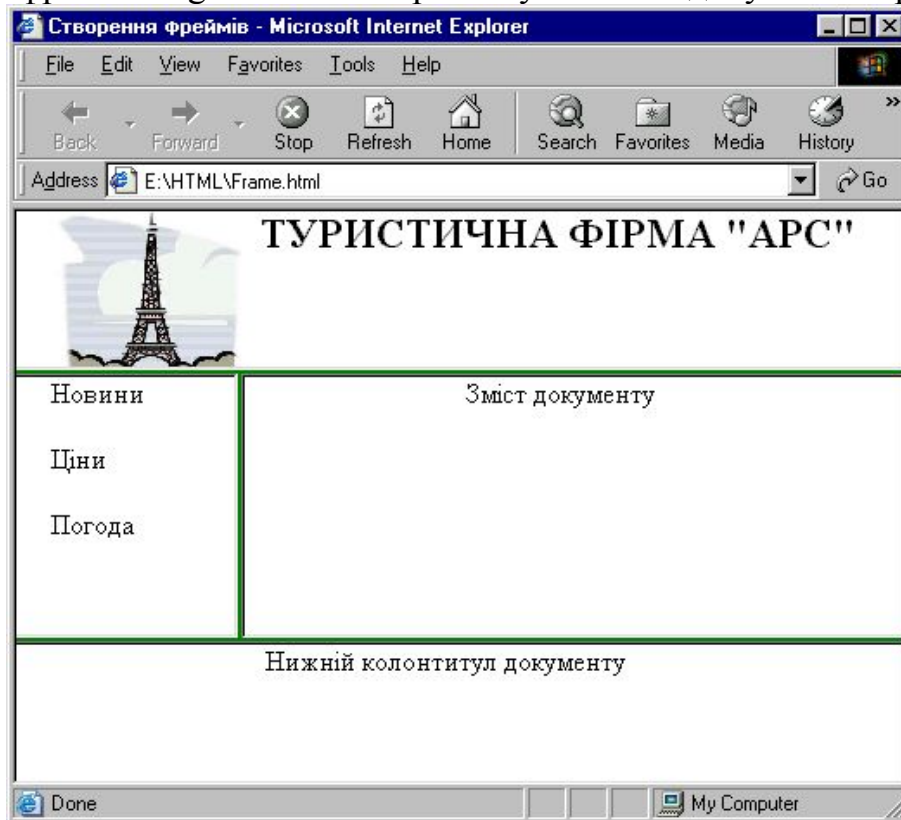
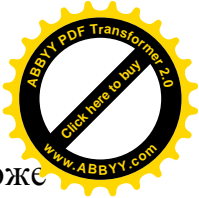


Рис.2.63. Відображення у браузері чотирьох фреймів, для яких визначено параметри *marginheight* та *marginwidth*, значення яких дорівнює 20 пікселів



9. Заборонити користувачам змінювати розмір фреймів, що може порушити структуру спроектованих нами фреймів. Реалізувати це за допомогою параметру `noresize` тегу `<frame>`, який не потребує ніяких значень.

10. Визначити взаємодію між фреймами.

У файлі `menu.html` створити гіперпосилання, перехід по яким буде завантажувати файл з іменем `example.html` у визначений фрейм. Для цього:

– Створити файл `example.html` в теці HTML та записати для нього HTML-код :

```
<html><head><title>example</title></head>
<body>
```

Текст документу

```
</body>
```

```
</html>
```

– Задати ім'я для фрейму, на який буде гіперпосилання – у нас це фрейм `content`. Скорегувати HTML-код для файлу `Frame.html` - фрейму `content` дати ім'я "A" :

```
<frameset rows="25%,50%,25%">
```

```
<frame src="logo.html" scrolling="no" frameborder=1 bordercolor="green"
marginheight="20" marginwidth="20">
```

```
<frameset cols="25%,*">
```

```
<frame src="menu.html" scrolling="no" frameborder=1
bordercolor="green"marginheight="20" marginwidth="20">
```

```
<frame src="content.html" frameborder=1 bordercolor="green"
marginheight="20" marginwidth="20" name="A">
```

```
</frameset>
```

```
<frame src="bottom.html" frameborder=1
```

```
bordercolor="green"marginheight="20" marginwidth="20">
```

```
</frameset>
```

– Скорегувати HTML-код для файлу `menu.html`: додати ще один пункт в меню (назвати його "Повідомлення"), визначити гіперпосилання на файл `example.html` :

```
<html>
```

```
<head><title>menu</title></head>
```

```
<body>
```

```
<a href="example.html"target="A">Новини</a><br><br>
```

```
<a href="example.html"target="_blank">Ціни</a><br><br>
```

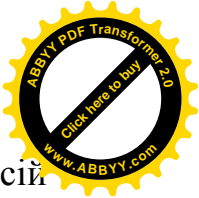
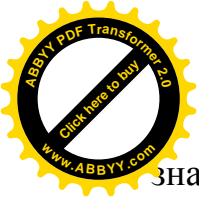
```
<a href="example.html"target="_top">Погода</a><br><br>
```

```
<a href="example.html"target="_self">Повідомлення</a>
```

```
</body>
```

```
</html>
```

Документ `menu.html` має чотири гіперпосилання на файл з іменем `example.html`, але з різним значенням параметру `target`. Перше гіперпосилання зі значенням `target="A"` буде завантажувати файл `example.html` у фрейм з іменем "A". Друге гіперпосилання зі значенням `target="_blank"` створить нове вікно без імені та завантажить туди необхідний документ. Третє гіперпосилання зі



значенням *target="_top"* завантажить документ у повне вікно замість всій фреймової структури. Четверте гіперпосилання зі значенням *target="_self"* завантажить документ у фрейм menu замість документу із гіперпосиланнями. Переглянемо оновлений файл Frame.html у вікні браузера (рис. 2.64)

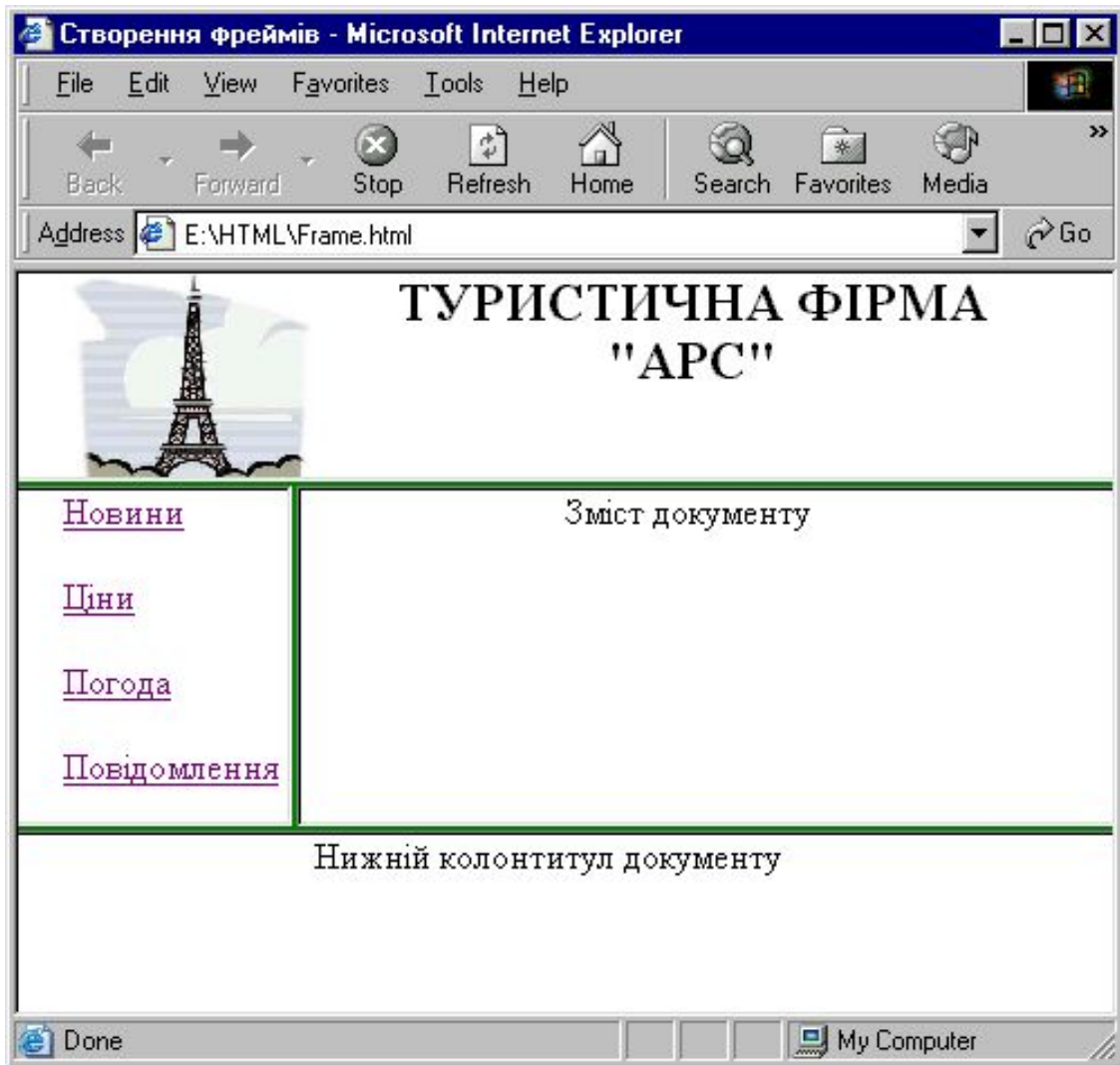


Рис. 2.64. Відображення оновленого документу Frame.html - у фреймі menu додано один пункт меню та визначені гіперпосилання

Переглянути у вікні браузера ситуації після реалізації чотирьох посилань на рис. 2.65, 2.66, 2.67.

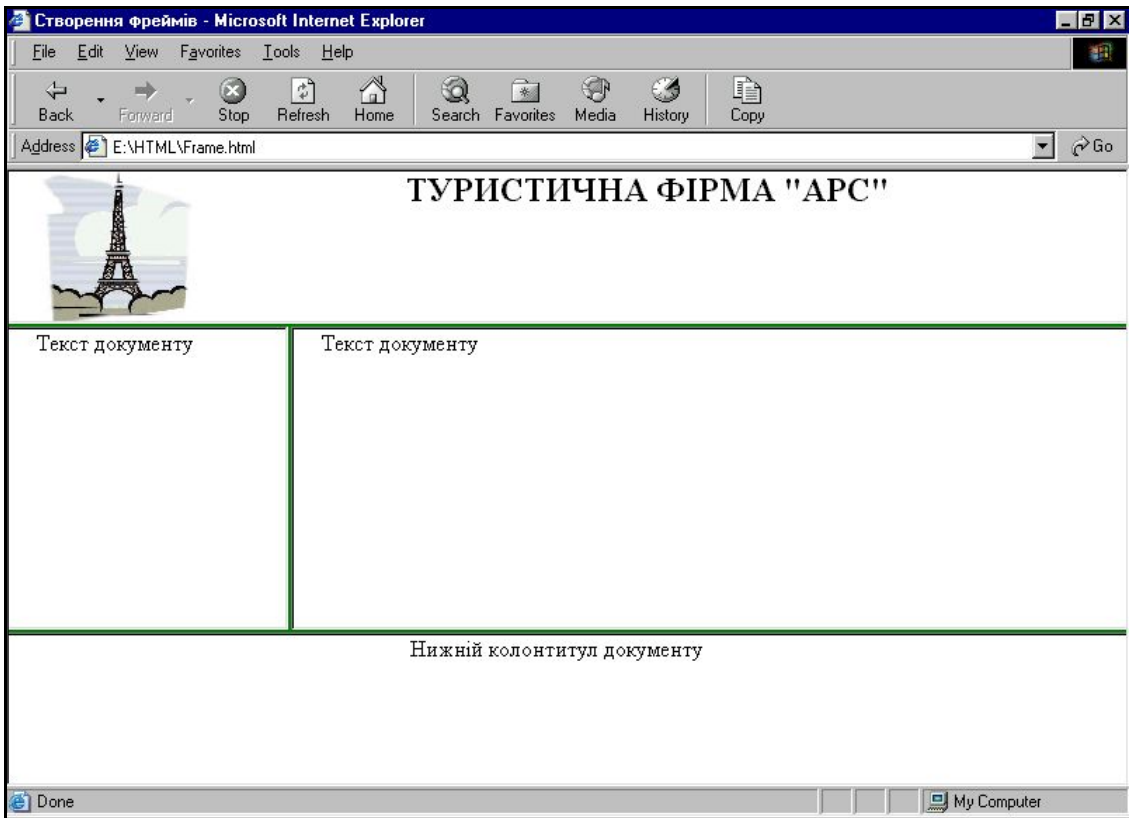


Рис. 2.65. Ситуація, отримана після послідовної реалізації першого та четвертого посилань

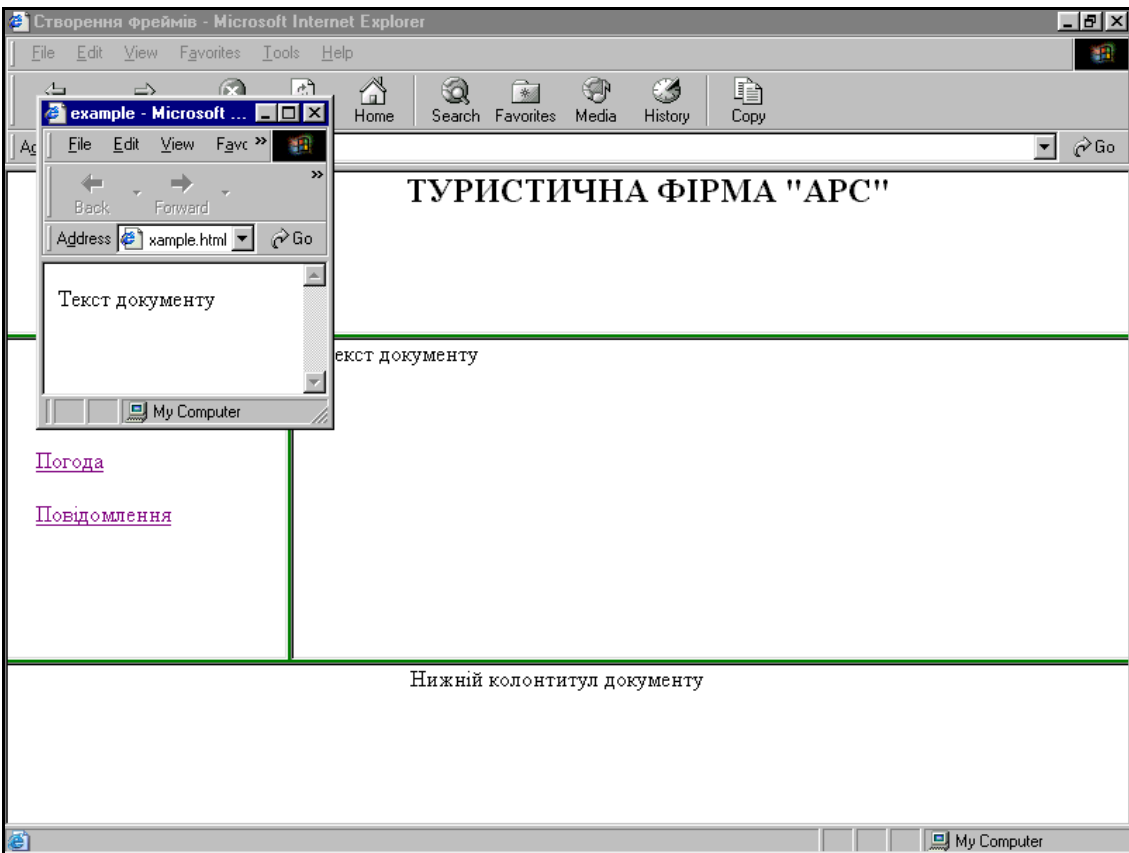


Рис. 2.66. Ситуація, отримана після реалізації другого посилання

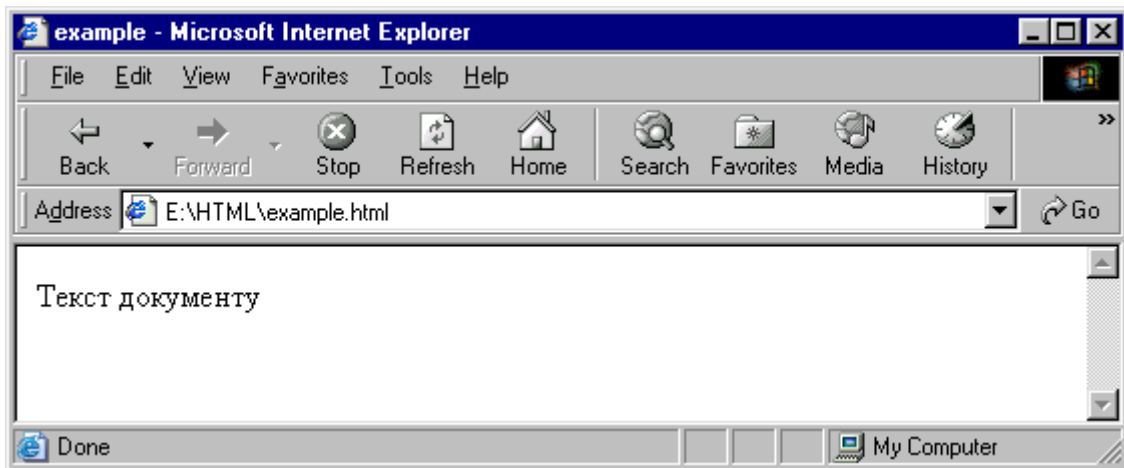


Рис. 2.67. Ситуація, отримана після реалізації третього посилання

Повторити друге посилання - файл example.html відкриється в ще одному, новому вікні браузера. Реалізувати третє посилання - нове вікно браузера не створюється, файл example.html відкривається в тому ж вікні де був фрейм. Повернутись до фреймової структури за допомогою кнопки Back. Звернути увагу, що кнопка Back активна на рис. 4.67 на відміну від рис 4.66.

11. За допомогою парного тегу `<iframe>` в документі logo1.html визначити плаваючий фрейм, в якому буде відображатись HTML-файл float.html. Для цього:

– В теці HTML створити файл float.html та записати в ньому наступний HTML-код:

```
<html>
<head>
  <title>float</title>
</head>
<body>
  <ul>
    <h2>Типи:</h2>
    <li>ОАЕ
    <li>Болгарія
    <li>Італія
    <li>Таїланд
  </ul>
</body>
</html>
```

– Створити файл logo1.html, який є копією файлу logo.html та буде розміщений в теці HTML. Скорегувати HTML-код документу logo1.html:

```
<html>
<head>
  <title>logo1</title>
</head>
<body>
  
```

```
<h2 align="center"> ТУРИСТИЧНА ФІРМА "АРС" </h2>
<iframe src="float.html" height="150" width="270" scrolling="yes"
align="right" valign="bottom" hspace="10" vspace="10">
Ваш браузер не дозволяє відображати плаваючі фрейми
</iframe>
```

```
</body>
```

```
</html>
```

Переглянути результат відображення цього HTML-коду у вікні браузера (рис. 2.68).

Якщо браузер не підтримує концепцію плаваючих фреймів, то в цьому випадку замість змісту документа float.html в ньому буде відображено текст "Ваш браузер не дозволяє відображати плаваючі фрейми".

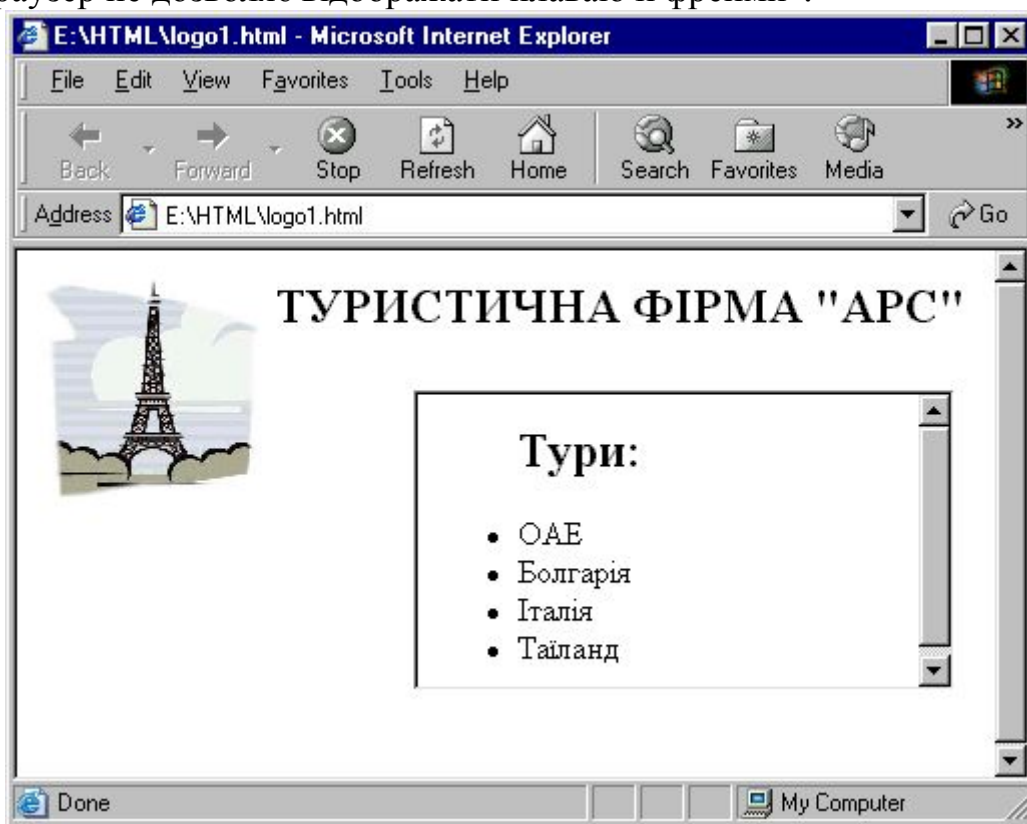


Рис. 2.68. Відображення плаваючого фрейму у вікні браузера

Контрольні питання:

1. Навіщо використовуються фрейми?
2. Як визначити кількість горизонтальних фреймів?
3. Як визначити кількість вертикальних фреймів?
4. Як визначити розмір фреймів в абсолютних величинах?
5. Як визначити розмір фреймів в відносних величинах?
6. Як встановити наявність границь між фреймами?
7. Як визначити який фрейм буде метою гіперпосилання?
8. Як визначити полосу прокрутки у фреймі?
9. Як встановити розміри "плаваючого" фрейму?
10. Як заборонити користувачеві змінювати розміри фреймів?



ЛІТЕРАТУРА

1. Антоненко В.М., Терейковський І.А., Терейковська Л.О. Основи Web-дизайну. Конспект лекцій. Навч. посібник. –К.: МПІ КСУ, 2005. – 116 с.
2. Антоненко В. М., Рогушина Ю.В. Сучасні інформаційні системи і технології. Навчальний посібник. - К.: КСУ МПІ, 2005. – 131 с.
3. Вандер Вер Э. JavaScript для чайников. К.: Издательский дом «Вильямс», 2001. – 304 с.
4. Гарнаев А., Гарнаев С. Web программирование на Java и JavaScript. - Санкт-Петербург: БХВ-Петербург, 2002.– 1040 с.
5. Кастаньетто Д. Профессиональное PHP программирование. СПб.: Символ-Плюс, 2001. - 912 с.
6. Колбери Р. Освой самостоятельно CGI за 24 часа. — М.: Издательский дом "Вильямс", 2001.— 368 с.
7. Костарев А. PHP в Web-дизайне. С.-Петербург.: ВHV, 2002. – 592 стр.
8. Матросов А., Сергеев А., Чаунин М. HTML 4.0. – Санкт-Петербург: БХВ-Петербург, 1999. - 672 с.
9. Паттерсон Л. и др. Использование HTML 4.0. - К.-М.-СПб.: Издат. Дом “Вильямс”. – 1998 – 384с.
10. Терейковський І.А. Підвищення ефективності функціонування корпоративних Web – сайтів. К.: Вісник КНУТД №4 2004 с.41-46.
11. Холл М., Браун Л. Программирование для Web. К.: Издательский дом «Вильямс», 2002. – 1264 с.
12. Антоненко В.М., Терейковський І.А., Терейковська Л.О. Основи Web-дизайну. Лабораторний практикум. Навчальний посібник. Частина 1–К.: ДРП та ПР КСУ, 2006. – 63 с.
13. Антоненко В.М., Терейковський І.А., Терейковська Л.О. Основи Web-дизайну. Лабораторний практикум. Навчальний посібник. Частина 2. – К.: ДРП та ПР КСУ, 2006. – 84 с.



**"ОСНОВИ WEB-ДИЗАЙНУ"
МЕТОДИЧНІ ВКАЗІВКИ І ЗАВДАННЯ
ДО САМОСТІЙНИХ І ЛАБОРАТОРНИХ РОБІТ**

(для студентів економічних спеціальностей)

Укладачі:

Бельков Дмитро Валерійович, к.т.н., доцент
Єдемська Євгенія Миколаївна, ст. викладач

Донецький національний технічний університет
83000, г.Донецьк-00, вул.Артема, 58