

Лабораторна робота №3

Тема. Проектування Web-документів. Сторінки списків та меню.

Мета. Створення Web-сторінок зі списками та меню Web-проекту. Вивчення **HTML** тегів та правил CSS роботи зі списками та посиланнями.

Теги: <a>, <link>, <nav>, , , , <dl>, <dt>, <dd>, <menu>, <command>, <menuitem>, <marquee>, <bdi> та <main>.

Властивості правил CSS: list-style-type, list-style-image, list-style-position, list-style, marker-offse, content, counter-reset, counter-increment.

Теги HTML.

Тег		Призначення
		Визначає неупорядкований список
		Визначає впорядкований список
		Визначає елемент списку
<dl>		Задає список визначень
<dt>		Задає термін визначення
<dd>		Задає опис визначення
<menu>		Визначає список
<menuitem>	5	Визначає елемент спимку в <menu>. Не багатьма браузерами.
<command>		Визначає командну кнопку в <menu>.
<a>		Визначає анкер
<link>		Визначає посилання на ресурс
<nav>	5	Визначає навігаційні посилання
<main>	5	Призначений для основного вмісту документа
<marquee>		Нестандартний тег. Визначає рядок символів, що біжить.
<bdi>	5	Ізолює фрагмент тексту, який може бути відформатований в іншому напрямку, тобто дозволяє браузерам коректно обробляти двонаправлений текст.

Завдання до лабораторної роботи №3

Розробити Web – сторінку виведення інформації у вигляді списків та Web – сторінку з меню для Web –проекту. На Web –сторінці студент виводить придуману інформацію фірму(підприємство) згідно варіану у вигляді списків. Списки пинні бути нумеровані, макеровані та із визначеннями. Сторінка меню повинна бути із гіпераосиланнями на поточну сторінку (внутрішнє посилання) та інші сторінки Web – проекту. У всіх розроблених раньше сторінах додати посилання на сторінку меню. При розробці сторінки студент може використовувати будь-які теги мови HTML та правила CSS. Для кожного студента, згідно варіанту, задається набір обов’язкових тегів та їх атрибутів, що повинні бути присутні на Web – сторінці. Задані атрибути можна замінити правилами CSS.

Варіанти завдань

Варіант	обов'язкові теги та їх атрибути
1	тег : атрибути - style, type="a", start, type="1"; тег : атрибути - start, value=square, compact, style ; тег : атрибути - value, class ; тег : <dd> атрибути - dir, title; тег : <dt> атрибути - title, style, id; тег : <command> атрибути - id, checked , class; тег : title; тег : <a> атрибути - rev , dir, class, target, href , title; тег : <link> атрибути - id, rel , sizes , type , style; тег : <nav> атрибути - class; тег : <marquee> атрибути - height , scrollamount ection , title, class, id; тег : <nobr> атрибути - id; class;
2	тег : атрибути - value, value=circle, id, type; тег : атрибути - class, type="1", compact, style, start; тег : атрибути - start, type, title, lang; тег : <dt> атрибути - class, id, style; тег : <dl> атрибути - id, dir; тег : <command> атрибути - style, checked ; тег : <dir> атрибути - title, , class; тег : <a> атрибути - href download , rel, target, accesskey , rev ; тег : <link> атрибути - media , charset , sizes , style, title; тег : <nav> атрибути - class; тег : <marquee> атрибути - bgcolor , scrolldelay , class, id, truespeed ection ; тег : <bdi> атрибути - id; тег : <main> атрибути - class;
3	тег : атрибути - class, style, value=disc, compact, value=circle; тег : атрибути - start, title, type, type="A" ; тег : атрибути - dir, value, compact; тег : <dl> атрибути - title, style ; тег : <dd> атрибути - id, class, style; тег : dir, title; тег : <menuitem> атрибути - style, id; тег : <a> атрибути - accesskey , href , class, coords , href type, title; тег : <link> атрибути - title, class, sizes , rel , media ; тег : <nav> атрибути - id; тег : <marquee> атрибути - scrollamount , style, scrolldelay , loop , height ; тег : <main> атрибути - id; class;
4	тег : атрибути - start, type="A", type, compact; тег : атрибути - value=disc, class , value=square, title; тег : атрибути - title, start, style, compact; тег : <dl> атрибути - id, class; тег : <dd> атрибути - title, class, style; тег : <menu> атрибути - dir, class; тег : <command> атрибути - style, disabled , checked ; тег : <a> атрибути - accesskey , title, coords , rev , shape, id, href ; тег : <link> атрибути - title, media , charset , id, class; тег : <nav> атрибути - id; тег : <marquee> атрибути - behavior , dir, style, hspace , scrollamount ; id; тег : <main> атрибути - id;
5	тег : атрибути - compact, class, title, start; тег : атрибути - compact, type="1", id, type ; тег : атрибути - class, id, title ; тег : <dl> атрибути - title, style; тег : <dd> атрибути - title , class; тег : <menu> атрибути - style, title, type ; тег : <menuitem> атрибути - dir, style, class; тег : <a> атрибути - download , name , id, href target, tabindex, accesskey ; тег : <link> атрибути - title , sizes , href , type ; тег : <nav> атрибути - id; тег : <marquee> атрибути - truespeed , id, class, vspace ection , style; тег : <bdi> атрибути - id; id;
6	тег : атрибути - type , value=circle, value=disc, start; тег : атрибути - compact, type="A", type="1", id, class; тег : атрибути - style , compact, lang; тег : <dt> атрибути - id, style; тег : <dl> атрибути - dir, id; тег : <menuitem> атрибути - style, class, lang; тег : <command> атрибути - style, checked , title; тег : <a> атрибути - shape, type, rel, target, id, style; тег : <link> атрибути - media , type , id, style; тег : <nav> атрибути - id; тег : <marquee> атрибути - id, truespeed , vspace , hspace ection ; тег : <nobr> атрибути - id; тег : <main> атрибути - class;
7	тег : атрибути - class, value=disc, compact, style, title; тег : атрибути - type="A", class, title, start; тег : атрибути - id, type, title, lang; тег : <dd> атрибути - dir, class; тег : <dt> атрибути - class, id, style; тег : <dir> атрибути - class, ; тег : <menuitem> атрибути - , , id; тег : <a> атрибути - download , accesskey , target, rel, type, tabindex; тег : <link> атрибути - id , sizes , charset , class; тег : <nav> атрибути - class; тег : <marquee> атрибути - title, class, vspace , id, bgcolor ; тег : <nobr> атрибути - id; id;
8	тег : атрибути - type, value=square, title, value=circle, value=disc; тег : атрибути - type="1", type, start, style, type="A"; тег : атрибути - class , value; тег : <dl> атрибути - class, title, id; тег : <dt> атрибути - title, class; тег : <command> атрибути - class, style, title; тег : <menuitem> атрибути - class, id, lang; тег : <a> атрибути - download , rel, style, class, name , id; тег : <link> атрибути - dir, rel , charset , id, style; тег : <nav> атрибути - class; тег :

	<marquee> атрибути - scrollamount , title, id , height , width ; тер : <main> атрибути - class; тер : <nobr> атрибути - id;
9	тер : атрибути - compact, value, title, class; тер : атрибути - type="1", style, id, class, lang; тер : атрибути - type, title, class, value; тер : <dl> атрибути - dir, id, title; тер : <dd> атрибути - title, style; тер : <menu> атрибути - class , id; тер : <menuitem> атрибути - id, class, ; тер : <a> атрибути - accesskey , coords , rev , type, id, download ; тер : <link> атрибути - sizes , title, href , class, media ; тер : <nav> атрибути - class; тер : <marquee> атрибути - truespeed , scrollamount , behavior , loop , hspace ; тер : <bdi> атрибути - class; тер : <nobr> атрибути - class;
10	тер : атрибути - start, type, compact, title, value; тер : атрибути - title, style, id, type="A", lang; тер : атрибути - style, start , value; тер : <dl> атрибути - style , title; тер : <dt> атрибути - class, id, lang; тер : <command> атрибути - dir, title, style; тер : <dir> атрибути - , id, ; тер : <a> атрибути - title, type, download , coords , class, accesskey , lang; тер : <link> атрибути - dir, type , id, sizes , title; тер : <nav> атрибути - class; тер : <marquee> атрибути - class, height , hspace , truespeed , loop , scrolldelay ; class; тер : <bdi> атрибути - class;
11	тер : атрибути - value=square, id, style, start; тер : атрибути - id, class, title, type="1"; тер : атрибути - compact, class, value, id; тер : <dd> атрибути - id, style; тер : <dt> атрибути - dir, style, title; тер : <dir> атрибути - style, dir; тер : <menuitem> атрибути - style, , title; тер : <a> атрибути - id, rel , title, shape, title, style; тер : <link> атрибути - sizes , class, type , charset , rel ; тер : <nav> атрибути - class; тер : <marquee> атрибути - dir, behavior , truespeed , class action , loop ; class; тер : <nobr> атрибути - class;
12	тер : атрибути - compact, type="1", start, value, type; тер : атрибути - value=disc, id, type, class, value=square; тер : атрибути - compact, id, value; тер : <dl> атрибути - title, id, lang; тер : <dt> атрибути - title, style ; тер : <menuitem> атрибути - id, class, ; тер : <command> атрибути - dir, disabled , class; тер : <a> атрибути - style, href , type, coords , class; тер : <link> атрибути - class, type , id, rel ; тер : <nav> атрибути - id; тер : <marquee> атрибути - truespeed , scrollamount , bgcolor , width , behavior , id; тер : <main> атрибути - class; тер : <nobr> атрибути - class;
13	тер : атрибути - title, start, style, type="a", value; тер : атрибути - dir, style, value, type, lang; тер : атрибути - dir, value, title; тер : <dt> атрибути - id , class; тер : <dd> атрибути - dir, id; тер : <dir> атрибути - , , class; тер : <menu> атрибути - type , class, label ; тер : <a> атрибути - target, coords , title, download , rel, title, accesskey ; тер : <link> атрибути - type , rel , class, style; тер : <nav> атрибути - id; тер : <marquee> атрибути - behavior , style, loop , title, class, hspace ; тер : <nobr> атрибути - id; id;
14	тер : атрибути - class, start, id, value, compact; тер : атрибути - title, value=square, class, id, lang; тер : атрибути - class, value, start ; тер : <dt> атрибути - dir, class, id; тер : <dl> атрибути - title, id ; тер : <dir> атрибути - title, id; тер : <menu> атрибути - type , id, class; тер : <a> атрибути - target, class, title, href dir, accesskey , lang; тер : <link> атрибути - dir, rel , id, title, type ; тер : <nav> атрибути - id; тер : <marquee> атрибути - truespeed , loop , scrolldelay , vspace action ; тер : <main> атрибути - class; class;
15	тер : атрибути - class, value=circle, compact, title; тер : атрибути - type="A", style , id; тер : атрибути - dir, style, start, compact; тер : <dt> атрибути - title, id, style; тер : <dd> атрибути - style, title, lang; тер : <command> атрибути - disabled , checked , id; тер : <dir> атрибути - id, class, ; тер : <a> атрибути - title, dir, rel, href , id, title; тер : <link> атрибути - charset , title, rel , style, href ; тер : <nav> атрибути - id; тер : <marquee> атрибути - width , bgcolor , truespeed , style, height ; тер : <main> атрибути - id; тер : <nobr> атрибути - id;
16	тер : атрибути - dir, class, compact, value=square, lang; тер : атрибути - dir, type, class, type="1", title; тер : атрибути - type, start, value, lang; тер : <dd> атрибути - dir, style, lang; тер : <dl> атрибути - title, class, style; тер : <menuitem> атрибути - style, , class; тер : <menu> атрибути - class, type ; тер : <a> атрибути - accesskey , name , class , target, download ; тер : <link> атрибути - id, type , media , rel , lang; тер : <nav> атрибути - id; тер : <marquee> атрибути - vspace , height , bgcolor , scrollamount , class, lang; class; тер : <nobr> атрибути - id;

17	тер : атрибути - value, class , start; тер : атрибути - value=circle, value, value=disc, style, id; тер : атрибути - dir, id, value; тер : <dl> атрибути - class , style; тер : <dd> атрибути - id, dir; тер : <dir> атрибути - , title, style; тер : <menuitem> атрибути - , style, class; тер : <a> атрибути - rel, shape, rev , href title, title; тер : <link> атрибути - href , title, id, sizes , lang; тер : <nav> атрибути - class; тер : <marquee> атрибути - dir ection , behavior , truespeed , bgcolor ; тер : <nobr> атрибути - class; тер : <bdi> атрибути - class;
18	тер : атрибути - title, compact, style, value=disc; тер : атрибути - title, type="1", type, type="a", lang; тер : атрибути - id, title, style, type; тер : <dd> атрибути - title, id; тер : <dt> атрибути - title , lang; тер : , title; тер : <command> атрибути - style , disabled ; тер : <a> атрибути - title, rel, type, shape, tabindex, href ; тер : <link> атрибути - type , sizes , rel , media ; тер : <nav> атрибути - class; тер : <marquee> атрибути - title, height , scrollldelay , hspace , class; тер : <nobr> атрибути - id; тер : <main> атрибути - id;
19	тер : атрибути - value=square, value, type, value=circle; тер : атрибути - type="A", style , value, compact; тер : атрибути - compact, class, id, style; тер : <dl> атрибути - style, class; тер : <dd> атрибути - title, class, id; тер : <command> атрибути - style, id, title; тер : <menu> атрибути - id, label ; тер : <a> атрибути - id , coords , href , type, title, shape; тер : <link> атрибути - id, class , sizes , media ; тер : <nav> атрибути - class; тер : <marquee> атрибути - id, scrollamount , width , bgcolor , lang; тер : <main> атрибути - id; class;
20	тер : атрибути - value=square, value, type, compact; тер : атрибути - title, style, type="1", id; тер : атрибути - type, compact, title, start; тер : <dt> атрибути - class , lang; тер : <dl> атрибути - class, title; тер : <dir> атрибути - class, title ; тер : <menu> атрибути - title, class, label ; тер : <a> атрибути - title, href shape, rev , style, title, href ; тер : <link> атрибути - rel , charset , href , title; тер : <nav> атрибути - id; тер : <marquee> атрибути - scrollldelay , truespeed , vspace , width , style, hspace ; тер : <main> атрибути - class; тер : <bdi> атрибути - class;
21	тер : атрибути - type="a", id, type="A" , value; тер : атрибути - title, value, value=disc, value=square, type; тер : атрибути - compact , class, start; тер : <dt> атрибути - title, style, id; тер : <dl> атрибути - style, class, lang; тер : <menu> атрибути - title, type , label ; тер : <command> атрибути - style, checked , disabled ; тер : <a> атрибути - name , style, accesskey , href type, rev , download ; тер : <link> атрибути - title, rel , class , lang; тер : <nav> атрибути - id; тер : <marquee> атрибути - width , scrollldelay , scrollamount , vspace , hspace , truespeed ; тер : <nobr> атрибути - id; id;
22	тер : атрибути - id, compact, value=disc, style, class; тер : атрибути - type="a", style, type="A", compact; тер : атрибути - class , type; тер : <dt> атрибути - title, dir; тер : <dl> атрибути - title, class, style; тер : <menuitem> атрибути - style, , ; тер : <menu> атрибути - id, label , type ; тер : <a> атрибути - class, tabindex, accesskey , title, rev , coords , shape; тер : <link> атрибути - style, media , rel , href ; тер : <nav> атрибути - class; тер : <marquee> атрибути - dir, class, scrollldelay , title ection , scrollamount ; тер : <main> атрибути - class; тер : <bdi> атрибути - id;
23	тер : атрибути - compact, type="a", type="A", type; тер : атрибути - value, type, class, style; тер : атрибути - value, class, start, style; тер : <dl> атрибути - dir, id; тер : <dd> атрибути - id, class, lang; тер : <menuitem> атрибути - , class; тер : <dir> атрибути - , style, title; тер : <a> атрибути - tabindex, class, shape, download , title, title; тер : <link> атрибути - rel , href , title, type , lang; тер : <nav> атрибути - id; тер : <marquee> атрибути - id, width , bgcolor , scrollamount , behavior ; class; тер : <main> атрибути - id;
24	тер : атрибути - title, value, type, value=disc; тер : атрибути - type="a", compact, id, value; тер : атрибути - compact, title, start, value; тер : <dl> атрибути - dir, id, title; тер : <dt> атрибути - title, class, lang; тер : <menu> атрибути - class, style ; тер : <command> атрибути - dir, title; тер : <a> атрибути - class, name , title, target, tabindex, id, rel; тер : <link> атрибути - sizes , type , href , class; тер : <nav> атрибути - class; тер : <marquee> атрибути - direction , height , bgcolor , scrollamount , title; тер : <nobr> атрибути - class; тер : <main> атрибути - class;
25	тер : атрибути - dir, value=circle, value=square, title, value; тер : атрибути - type="1", value, type="a", id, class; тер : атрибути - value, start ,

	compact; тер : <dd> атрибути - title, id; тер : <dl> атрибути - style, title, lang; тер : <div> атрибути - , ; тер : <menuitem> атрибути - title, ; тер : <a> атрибути - href download , id, accesskey , title, coords ; тер : <link> атрибути - media , title, rel , href ; тер : <nav> атрибути - class; тер : <marquee> атрибути - truespeed , class, loop , height , bgcolor , id; тер : <main> атрибути - class; тер : <nobr> атрибути - class;
26	тер : атрибути - value=circle, title, style, id; тер : атрибути - id, class, type="A", start; тер : атрибути - title, id, type, compact; тер : <dl> атрибути - id, title ; тер : <dt> атрибути - style, class, lang; тер : style, ; тер : <menu> атрибути - label , type , title; тер : <a> атрибути - href style, id, download , class, href ; тер : <link> атрибути - media , charset , type , href ; тер : <nav> атрибути - class; тер : <marquee> атрибути - scrolldelay , title, height , vspace ; тер : <bdi> атрибути - class; тер : <nobr> атрибути - id;
27	тер : атрибути - type="A", style , value; тер : атрибути - type, style, value=square, value=disc, lang; тер : атрибути - style , compact, start; тер : <dd> атрибути - dir, title, id; тер : <dt> атрибути - dir, title; тер : <div> атрибути - , title, id; тер : <command> атрибути - class, disabled , style; тер : <a> атрибути - title, title, shape, accesskey , rev , id; тер : <link> атрибути - type , charset , id, style; тер : <nav> атрибути - id; тер : <marquee> атрибути - style, height , width , vspace ; тер : <nobr> атрибути - class; id;
28	тер : атрибути - title, value=circle, class, id; тер : атрибути - style, type="1", class, type, title; тер : атрибути - title, class, value, style; тер : <dl> атрибути - class, title; тер : <dd> атрибути - id ; тер : <menuitem> атрибути - , ; тер : <command> атрибути - style, id ; тер : <a> атрибути - href , title, target, rev , accesskey , tabindex ; тер : <link> атрибути - id, style, title, media , rel ; тер : <nav> атрибути - id; тер : <marquee> атрибути - hspace , bgcolor , truespeed , behavior , width , height ; class; тер : <nobr> атрибути - class;
29	тер : атрибути - value=disc, value=circle, value=square, start; тер : атрибути - type="A" , title, value, type="a"; тер : атрибути - compact, class, id; тер : <dt> атрибути - style, id ; тер : <dl> атрибути - id, class, lang; тер : <menu> атрибути - title, type , id; тер : title, style; тер : <a> атрибути - shape, target, tabindex, href type, coords ; тер : <link> атрибути - charset , type , href , media ; тер : <nav> атрибути - class; тер : <marquee> атрибути - hspace , bgcolor , title, scrolldelay , style; тер : <nobr> атрибути - id; тер : <main> атрибути - id;
30	тер : атрибути - class, style, value=disc, value=square, start; тер : атрибути - type="a", type, title, compact, value; тер : атрибути - class , start; тер : <dl> атрибути - dir, title, class; тер : <dd> атрибути - style, title; тер : <menu> атрибути - type , style; тер : <command> атрибути - class, title, id; тер : <a> атрибути - class, download , shape, coords , href , rev ; тер : <link> атрибути - href , sizes , media , style, class; тер : <nav> атрибути - id; тер : <marquee> атрибути - scrollamount , class, scrolldelay , width , loop ; тер : <main> атрибути - id; тер : <nobr> атрибути - id;
31	тер : атрибути - type, id, style, title, start; тер : атрибути - title, value, value=square, type; тер : атрибути - compact, title, class; тер : <dt> атрибути - title, class ; тер : <dl> атрибути - style, title, id; тер : <command> атрибути - checked , disabled , class; тер : <menuitem> атрибути - style, id ; тер : <a> атрибути - class, download , rev , tabindex, coords , href title; тер : <link> атрибути - class, charset , media , sizes ; тер : <nav> атрибути - class; тер : <marquee> атрибути - scrollamount , behavior , width , class, vspace , bgcolor ; тер : <nobr> атрибути - class; тер : <main> атрибути - class;

Приклад сторінки.

```

<!DOCTYPE html>

<html lang=uk>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <meta name="Author" lang="uk" content="Гаштемеленко Яваскриптик Перлович">
  <meta name="Keywords" content="Комп'ютерні мережі, списки, новини, компанія ">
  <style>
    .term {

```

```

        color: green
    }

    .term2 {
        color: blue
    }

    img {
        border : 0
    }

</style>

<title>Приклад. Сторінка списки </title>
</head>
<body>
<h1><b>Модуль 1. Комп'ютерні мережі</b></h1>
<marquee>    Комп'ютерні мережі    </marquee>
<h2>
    <br><span class="term"> 1.1. Введення в дисципліну "КТС".</span>
</h2>
<br><br>Перша глобальна мережа - Arpanet (Міністерство оборони США).
<br>Схема мережі "клієнт - сервер":
<br><br>
<br><br>У моделі "клієнт - сервер" розрізняють запити й відповіді.
<br><br>

<br>
<ol>
    <li>
        Застосування КС.
        <br>До появи мереж робота користувачів велася ізолированно, тобто для передачі даних
        використовувалися носії інформації.
    <li>
        Ціль побудови мереж.
        <br>Спільне використання фізичних ресурсів (принтер, факс, модем); користувачеві надається
        доступ до ПЗ, устаткуванню, даним, незалежно від фізичного розташування ресурсів і користувача.
    <li>Спільне використання інформації БД в електронному виді (фінанси, податки, рахунки, кадри).
    <li>
        <span class="term">Інформаційна система компанії</span> - сукупність однієї або декількох БД і
        деякої кількості службовців, яким передається дистанційний доступ до даних.
        <br><span class="term">Сервер</span> - сховище даних.
        <br><span class="term">Клієнти</span> - машини службовців ( у межах офісу або вилученої машини).
        <br>Клієнти й сервер об'єднані в мережу. Така система називається клієнт - серверною моделлю -
        це основа побудови мережі.
        <br>Обмін інформацією відбувається в такий спосіб:
        <ul>
            <li>клієнт надсилає запит на сервер і очікує відповіді;
            <li>сервер, одержавши запит, виконує дії, шукає дані, відсилає відповідь.
        </ul>
    <li><span class="term">Електронна пошта (e-mail)</span> - комунікаційне середовище для службовців.
    <li>Спільна робота над тими самими документами, звітами, проектами.
    <li><span class="term">Відеоконференції</span> - зустрічі співрозмовників з різних куточків миру
        (віртуально).
    <li>Можливість електронного ділового спілкування з іншими компаніями (постачальник - клієнт).
    <li><span class="term"> Інтернет-Комерція</span> - електронний бізнес.
</ol>
</h2>
<span class="h1">1.2. Використання мереж частками особами</span>
</h2>
<br><br>Домашнє використання КТС:
<ul>
    <li>
        доступ до вилученої інформації
        <ul>
            <li>одержання електронних газет
            <li>доступ до онлайн-бібліотек
        </ul>
    <li>
        спілкування
        <ul>
            <li>e-mail
            <li>чати (chat - система діалогових повідомлень у режимі онлайн)
            <li>всесвітні конференції (обговорення всіляких тем, але не в реальному часі)
            <li>
                технології рівнорангових (однорангових) мереж (peer-to-peer). Відрізняються від моделі "
                клієнт-сервер" тим, що кожен може спілкуватися один з одним. Кожен може дати в загальне користування свої
                дані й одержати доступ до даних іншого користувача.
                <br><br>
            <br>
            <li>ip-телефонія
            <li>інтернет-радіо
            <li>електронні азартні ігри
            <li>електронні освітні програми
            <li>дистанційне навчання
        </ul>
    <li>
        інтерактивні розваги
        <ul>
            <li>мережні ігри
            <li>інтерактивне телебачення й відео
        </ul>
    </li>
</ul>

```

```

<li>
    електронний бізнес
    <ul>
        <li>електронні магазини
        <li>електронні "блошині" ринки
    </ul>
</li>
</ul>
<h3>Типи серверів: </h3>
<dl>
    <dt>
        <span class="term2"> Сервер друку </span>.
    </dt>
    <dd> Приєднання принтера до мережі через спеціалізований вузол обробки завдань на друк </dd>
    <dt>
        <span class="term2">Сервер баз даних</span>.
    </dt>
    <dd> Спільно використовувана база даних. </dd>
    <dt>
        <span class="term2"> Файл-Сервер</span>.
    </dt>
    <dd> Сховище даних.</dd>
    <dt>
        <span class="term2"> Комунікаційний сервер</span>.
    </dt>
    <dd> Управляє доступом до вилучених ресурсів; забезпечує канал зв'язку із глобальною обчислювальною мережею. </dd>
    <dt>
        <span class="term2"> Сервер додатків</span>. </dt>
    <dd> Виконуються спеціальні обчислювальні завдання (обробка графіки).</dd>
    <dt>
        <span class="term2">Веб-сервер</span>.
    </dt>
    <dd> Забезпечує доступ до веб-сторінок. </dd>
</dl>
<A id="1.3"></A>
<h2><span class="h1">1.3. Основні мережні поняття</span> </h2>
<br><br><span class="term">Комп'ютерна мережа</span> - сукупність вузлів (ПК, терміналів, периферійних пристроїв), що мають можливість інформаційної взаємодії один з одним за допомогою спеціального комунікаційного устаткування й програмного забезпечення (ПО).
<br>Розміри мереж:
<ol>
    <li>Пари з'єднаних між собою комп'ютерів.
    <li>Мільйони комп'ютерів, розкиданих по усьому світу.
</ol>
По ширині охопту прийнятий розподіл мереж на категорії:
<ol>
    <li><span class="term2"><abbr>ЛОМ</abbr></span> - <abbr>LAN</abbr> (Local-Area Network). Поеднує ПК, розташовані в обмеженому просторі. Для них необхідна прокладка спеціалізованої кабельної системи. Положення можливих точок підключення абонентів обмежене цією кабельною системою.
    <li>
        <span class="term2"><abbr>ГОМ</abbr></span> - <abbr>GAN</abbr> (Global-Area Network).
        Прикладом глобальної мережі може бути мережа Internet.
        <br>Між <abbr>ЛОМ i </abbr>ГОМ перебувають наступні категорії мереж:
        <ul>
            <li><span class="term2">Кампусная мережа</span> - <abbr>CAN</abbr> (Campus-Area Network). Поеднує ЛОМ у близько розташованих будинках.
            <li><span class="term2">Мережа міського масштабу</span> - <abbr>MAN</abbr> (Metropolitan-Area Network). Об'єднання розширених локальних мереж підприємств між собою в межах району міста.
            <li><span class="term2">Широкомасштабна мережа</span> - <abbr>WAN</abbr> (Wide-Area Network). Об'єднання мереж у межах великого мегаполіса.
        </ul>
    </li>
</ol>
Для більших мереж також установлюються спеціальні провідні або бездротові лінії зв'язку, або ж використовується інфраструктура існуючих суспільних засобів зв'язку ( телефонна мережа). В останньому випадку абоненти комп'ютерної мережі можуть підключатися до мережі у відносно довільних крапках, охоплених мережею телефонії, ISDN (цифрова мережа з комплексними послугами) або кабельного телебачення.
<br><span class="term">Інтранет (Intranet)</span> - позначає внутрішню мережу організацій. Для неї важливі два моменти:
<ol>
    <li>Ізоляція (або захист) внутрішньої мережі від зовнішньої.
    <li>Забезпечення виходу на зовнішню мережу.
</ol>
</body>
</html>

```

Додаток.

Списки

1. Загальний опис

HTML підтримує три способи зберігання й відображення *списків*. Любою список складається з одного або декількох *елементів списків*. Списки підрозділяються на:

- маркіровані (неупорядковані) списки ``;
- нумеровані (упорядковані) списки ``;
- списки визначень.

Тільки що наведений список є маркірованим і виглядає мовою HTML так:

```
<UL>
  <LI>маркіровані (неупорядковані) списки;</LI>
  <LI>нумеровані (упорядковані) списки;</LI>
  <LI>списки визначень</LI>
</UL>
```

Нумерований список виглядає аналогічно, але його елементи нумеруються:

1. перший елемент списку;
2. другий елемент списку;
3. третій елемент списку.

Мовою HTML це записується так:

```
<OL>
  <LI>перший елемент списку;</LI>
  <LI>другий елемент списку;</LI>
  <LI>третій елемент списку</LI>
</OL>
```

Нарешті, списки визначень складаються з пар термін/визначення, хоча їхнє застосування набагато ширше. Приклад використання списку визначень для складання театрального репертуару:

1 липня
А. К. Толстой. *Цар Федір Иоаннович*
4 липня
У. Теккерей. *Ярмарок марносластва*
9 липня
А. Островський. *Вовки й вівці*

Мовою HTML це записується так:

```
<DL>
  <DT><STRONG>1 липня</STRONG></DT>
  <DD>А. К. Толстой. <EM>Цар Федір Иоаннович</EM></DD>
  <DT><STRONG>4 липня</STRONG></DT>
  <DD>У. Теккерей. <EM>Ярмарок марносластва</EM></DD>
  <DT><STRONG>9 липня</STRONG></DT>
  <DD>А. Островський. <EM>Вовки й вівці</EM></DD>
</DL>
```

1. Списки можуть вкладатися друг у друга, причому допускається вкладення списків одного типу в списки іншого типу.

Спосіб відображення списків залежить від оглядача й може бути уточнений за допомогою завдання [стиля списку](#).

2. Маркіровані списки: елемент UL

Синтаксис: `...` (блоковий елемент)

Атрибути: `id`, `class`, `style`, `title`, `lang`, `dir`, `события`

Підтримка: Повна відповідність стандарту (5.0+)

Атрибут `dir` ігнорується (4.0+)

Елемент **UL** (unordered list) визначає *маркірований список*. Його вмістом повинні бути один або кілька елементів **LI**. Оглядачі звичайно відображають елементи маркірованого списку з відступом вправо й з маркером перед початком елемента (див. приклади,

наведені вище). Властивість стилів **list-style** дозволяє авторам змінювати вид маркерів, забороняти їхній вивід і т.п.

3. Нумеровані списки: елемент OL

Синтаксис: `...` (блоковий елемент)

Атрибути: `id`, `class`, `style`, `title`, `lang`, `dir`, `события`

Елемент **OL** (ordered list) визначає *нумерований список*. Його вмістом повинні бути один або кілька елементів **LI**. Оглядачі звичайно відображають елементи нумерованого списку з відступом вправо й з номером перед початком елемента (див. приклади, наведені вище). Властивість стилів **list-style** дозволяє авторам змінювати спосіб нумерації елементів списку.

4. Елементи списків: елемент LI

Синтаксис: `...` (блоковий елемент, кінцевий тег необов'язковий)

Атрибути: `id`, `class`, `style`, `title`, `lang`, `dir`, `события`

Елемент **LI** (list item) визначає *елемент* маркірованого або нумерованого *списку*. Він може містити в собі інші блокові елементи, у т.ч. й елементи **OL** і **UL**, дозволяючи тим самим створювати багаторівневі вкладені списки. Спосіб його відображення залежить від типу списку й може бути змінений властивістю стилів **list-style**.

5. Списки визначень: елементи DL, DT й DD

Синтаксис: `<DL>...</DL>` (блоковий елемент)

Атрибути: `id`, `class`, `style`, `title`, `lang`, `dir`, `события`

Синтаксис: `<DT>...</DT>` (блоковий елемент, кінцевий тег необов'язковий)

Атрибути: `id`, `class`, `style`, `title`, `lang`, `dir`, `события`

Синтаксис: `<DD>...</DD>` (блоковий елемент, кінцевий тег необов'язковий)

Атрибути: `id`, `class`, `style`, `title`, `lang`, `dir`, `события`

Елемент **DL** (definition list) визначає *список визначень*. Він відрізняється від інших типів списків тим, що кожен елемент цього списку представляється двома елементами HTML: елементом **DT** (definition term), що містить *ім'я обумовленого терміна*, і елементом **DD** (definition description), що містить *визначення цього терміна* (див. приклади, наведені вище). Елемент **DT** може містити будь-які текстові (але не блокові) елементи. Елемент **DD** може містити як текстові, так і блокові елементи, дозволяючи тим самим створювати багаторівневі вкладені списки.

Стандарт не виключає можливості давати кілька визначень одного терміна або одне визначення декільком термінам, як показано в наступному прикладі:

```
<DL>
  <DT>Uniform Resource Identifier</DT>
  <DT>Уніфікований ідентифікатор ресурсу</DT>
  <DD>Стандартизований рядок, що вказує на ресурс Інтернету,
    такий як HTML-документ або дисковий файл</DD>
</DL>
```

Цей приклад буде відображатися так:

Uniform Resource Identifier

Уніфікований ідентифікатор ресурсу

Стандартизований рядок, що вказує на ресурс Інтернету, такий як HTML-документ або дисковий файл.

Як відзначалося вище, списки визначень мають досить широке коло застосувань. Одним з них є подання діалогів, у яких елемент **DT** містить ім'я мовця, а **DD** - його або її слова. Приклад такого діалогу з "Вишневого саду" А. П. Чехова:

```
<DL>
  <DT>Фірс</DT>
  <DD>Перед нещастям теж була: і сова кричала, і самовар гудів бесперечь.</DD>
  <DT>Гаїв</DT>
  <DD>Перед яким нещастям?</DD>
  <DT>Фірс</DT>
  <DD>Перед волею</DD>
</DL>
```

Цей діалог буде відображатися так:

Фірс

Перед нещастям теж була: і сова кричала, і самовар гудів бесперечь.

Гаїв

Перед яким нещастям?

Фірс

Перед волею.

6. Гіперпосилання й закладки: елемент A

Хоча HTML містить велика кількість засобів для форматування тексту й структурування документів, його основною особливістю є можливість створення *гіпертекстових документів*. Гіпертекстовими є документи, які містять *гіперпосилання* на інші ресурси Веб.

Гіперпосилання — це зв'язування між двома *анкерами*: анкер джерела, що *втримується* в даному HTML-документі, *вказує* на анкер *мети*, що може бути HTML-документом або *закладкою* в ньому, а також графічним файлом, *аудіофайлом*, *відеокліпом* або *програмою*.

Для створення анкерів у мові HTML використовується елемент **A**.

Синтаксис: <A>... (текстовий елемент)

Атрибути: id, class, style, title, lang, dir, события

name = CDATA (назва закладки)

href = URI (URI посилання)

rel = типы ссылок (пряме посилання)

rev = типы ссылок (зворотне посилання)

type = тип файла (тип файлу, на який вказує посилання)

target = фрейм (фрейм для відображення посилання)

hreflang = код языка (мова посилання)

charset = кодировка (кодування посилання)

accesskey = символ (ім'я клавіші швидкого доступу)

tabindex = число (номер у порядку натискання табуляції)

shape = rect | circle | poly | default (область у карті посилань)

coords = координати (координати в карті посилань)

onfocus = сценарий (при одержанні елементом фокуса)

onblur = сценарий (при втраті елементом фокуса)

Елемент **A** (anchor) визначає положення анкера в документі. Він застосовується для створення *закладок* і для створення *гіперпосилань*. Його вмістом можуть бути будь-які текстові елементи, крім **A**.

Для створення закладок (тобто анкерів мети) використовується атрибут **name**.

Наприклад, що впливає елемент у документі mydoc.html визначає закладку myheading:

```
<H1><A name="myheading">Мій заголовок</A></H1>
```

Для створення гіперпосилання на неї з того ж документа використовується анкер:

```
<A href="#myheading">...</A>
```

а з іншого документа — анкер:

```
<A href="mydoc.html#myheading">...</A>
```

Примітки.

1. Елемент **A** не може бути вкладений в інший елемент **A**, тому стежите, щоб анкери закладок не містили анкерів гіперпосилань. У запобіганні цього можна використати в одному елементі **A** й атрибут **name**, і атрибут **href**.
2. Стандарт HTML рекомендує використати замість атрибута **name** атрибут **id**. Сучасні оглядачі підтримують цю можливість, однак це може викликати труднощі в динамічних сторінках через *несовместимости обозревателей*.

Для створення гіперпосилань (тобто анкерів джерела) використовується атрибут **href**. Його значенням повинен бути URI ресурсу, на який вказує гіперпосилання. Анкер гіперпосилання може мати наступні необов'язкові атрибути:

- Атрибут **title** задає короткий опис анкера мети й відображається як "підказка".
- Атрибут **target** використовується тільки у фреймових документах і задає імя фрейма, у якому повинна відобразитися мета гіперпосилання.
- Атрибути **hreflang** й **charset** вказують на язык і кодировку посилання.

- Атрибут **accesskey** задає символ Unicode як клавіша швидкого переходу по гіперпосиланню.
- Значенням атрибута **tabindex** повинне бути ціле число в діапазоні від 0 до 32767. Це число задає номер даного гіперпосилання при русі по Веб-сторінці натисканнями клавіші Tab. Якщо **tabindex** дорівнює 0 або не задано, то даний елемент одержить фокус після всіх елементів з позитивним значенням **tabindex**. При збігу значень **tabindex** у декількох елементів першим одержує фокус той, котрий раніше зустрічається в документі.
- Атрибути **rel** й **rev** мають те ж призначення, що й в елемента **LINK**.
- Атрибути **shape** й **coord** використовуються тільки в картах посилань й описані в [п. 6.6.1](#).

Приклади:

```
<lt href="album.html">Мій фотоальбом</A>
<A href="../images/me.jpg">Моя фотографія</A>
<A href="sounds/yester.mid" type="audio/midi" accesskey="Y">Yesterday
(MIDI-файл)</A>
<A href="section2.html" title="Таблиці стилів" rel="next"
target="_blank">Розділ 2</A>
<A href="mailto:myname@mail.com" title="Повідомите Вашу
думку">myname@mail.com</A>
<A href="http://babel.alis.com:8080/langues/iso639.zh.htm" charset="big5"
hreflang="zh">ISO 639 (китайська мова)</A>
```

Крім базових подій, елемент **A** допускає обробку ще двох подій:

- **onfocus** — відбувається, коли гіперпосилання одержує фокус;
- **onblur** — відбувається, коли гіперпосилання губить фокус.

2.11.5. Списки CSS

2.11.5.1. Введення в списки

Елементи списків створюються присвоюванням значення *list-item* властивості **display**. Як й у більшості загальному випадку маркерів, вони відображаються у вигляді двох прямокутників: головного прямокутника для вмісту елемента й необов'язкового прямокутника маркера списку. Властивості списків дозволяють задати тип цього маркера і його положення щодо головного прямокутника. Відзначимо, що при відображенні списків тло елемента поширюється тільки на головний прямокутник; прямокутник маркера завжди прозорий.

2.11.5.2. Тип маркера списку: властивість **list-style-type**

Синтаксис: `list-style-type: disc | circle | square | decimal | decimal-leading-zero | lower-roman | upper-roman | lower-greek | lower-alpha | lower-latin | upper-alpha | upper-latin | hebrew | armenian | georgian | cjk-ideographic | hiragana | katakana | hiragana-iroha | katakana-iroha | none | inherit`

Начально: `disc`

Застосовно: до елементів типу `display: list-item`

Наслідувано: да

Відсотки: не використовуються

Пристрою: [визуальные](#)

Підтримка: Підтримуються тільки `disc`, `circle`, `square`, `none`, `decimal`, `lower-roman`, `upper-roman`, `lower-alpha`, `upper-alpha` (4.0+)

Підтримуються тільки `disc`, `circle`, `square`, `none`, `decimal`, `lower-roman`, `upper-roman`, `lower-alpha`, `upper-alpha` (4.0+)

Властивість **list-style-type** задає *тип маркера списку*. Воно може приймати наступні значення:

<i>disc</i>	Чорний кружок (залежить від оглядача)
<i>circle</i>	Світлий кружок (залежить від оглядача)
<i>square</i>	Чорний квадратик (залежить від оглядача)
<i>decimal</i>	Десяткові числа (1, 2, 3, ...)
<i>decimal-leading-zero</i>	Десяткові числа із провідним нулем (01, 02, 03, ... 98, 99)
<i>lower-roman</i>	Рядкові римські числа (i, ii, iii, iv, v, ...)
<i>upper-roman</i>	Прописні римські числа (I, II, III, IV, V, ...)
<i>lower-greek</i>	Рядкові грецькі букви (α , β , γ , ..., ω)
<i>lower-alpha</i>	Рядкові латинські букви (a, b, c, ..., z)
<i>lower-latin</i>	Рядкові латинські букви (a, b, c, ..., z)
<i>upper-alpha</i>	Прописні латинські букви (A, B, C, ..., Z)
<i>upper-latin</i>	Прописні латинські букви (A, B, C, ..., Z)
<i>hebrew</i>	Традиційна івритська нумерація
<i>armenian</i>	Традиційна вірменська нумерація
<i>georgian</i>	Традиційна грузинська нумерація (an, ban, gan, ..., he, tan, in, in-an, ...)
<i>cjk-ideographic</i>	Далекосхідна ідеографічна нумерація
<i>hiragana</i>	Японська нумерація абеткою хирагана (a, i, u, e, o, ka, ki, ...)
<i>katakana</i>	Японська нумерація абеткою катакана (A, I, U, E, O, KA, KI, ...)
<i>hiragana-iroha</i>	Японська нумерація хирагана-іроха (i, ro, ha, ni, ho, he, to, ...)
<i>katakana-iroha</i>	Японська нумерація катакана-іроха (I, RO, HA, NI, HO, HE, TO, ...)

Специфікація CSS не визначає, які символи повинні використатися як маркери списку, коли заданий алфавіт буде вичерпаний. Наприклад, 27-й елемент списку типу *lower-alpha* не визначений. Тому алфавітні типи маркерів повинні використатися тільки для коротких списків заздалегідь відомого розміру. Якщо оглядач не підтримує заданий тип маркера, то він повинен замінятися на *decimal*.

Ця властивість застосовується до списку тільки тоді, коли значенням властивості **list-style-image** є *none* або зазначений у ньому графічний образ не може бути завантажений.

Наступний приклад нумерує елементи нумерованого списку рядковими римськими числівниками:

```
OL { list-style-type: lower-roman }
```

2.11.5.3. Графічний маркер списку: властивість **list-style-image**

Синтаксис: `list-style-image: <uri> | none | inherit`

Начально: *none*

Застосовно: до елементів типу `display: list-item`

Наслідувано: да

Відсотки: не використовуються

Пристрою: [визуальні](#)

Підтримка: Відповідає стандарту (4.0+)

Не підтримується

Властивість **list-style-image** задає *графічний маркер списку* вказівкою на URI відповідного графічного образа. Якщо воно дорівнює *none*, то тип маркера списку задається властивістю **list-style-type**.

Наступний приклад поміщає в початок кожного елемента маркірованого списку графічний образ "ellipse.gif":

```
UL { list-style-image: url("http://mylibrary.com/ellipse.gif") }
```

2.11.5.4. Позиція маркера списку: властивість **list-style-position**

Синтаксис: `list-style-position: inside | outside | inherit`

Начально: *outside*

Застосовно: до елементів типу `display: list-item`

Наслідувано: да

Відсотки: не використовуються

Пристрою: [визуальные](#)

Підтримка: Відповідає стандарту (4.0+)

Не підтримується

Властивість **list-style-position** задає положення маркера списку щодо елемента списку. Воно може приймати наступні значення:

inside

Прямокутник маркера стає першим текстовим блоком головного прямокутника.

outside

Прямокутник маркера розміщується поза головним прямокутником.

Наприклад, що впливає фрагмент HTML-документа

```
<UL style="list-style-type: square; list-style-position: inside; width: 40%">
  <LI>Перший елемент списку розташований таким от образом</LI>
  <LI>Другий елемент списку розташовується так же.</LI>
</UL>
```

буде відображатися так (Netscape Navigator 4.x відображає цей приклад невірно!):

- Перший елемент списку розташований у такий от спосіб.
- Другий елемент списку розташовується так само.

2.11.5.5. Завдання властивостей списку: властивість list-style

Синтаксис: `list-style: [<list-style-type>|<list-style-image>|<list-style-position>] | inherit`

Начально: не визначено для скорочень

Застосовно: до елементів типу `display: list-item`

Наслідувано: да

Відсотки: не використовуються

Пристрою: [визуальные](#)

Підтримка: Відповідає стандарту (4.0+)

Не підтримується

Властивість **list-style** є скороченням для властивостей **list-style-type**, **list-style-image** і **list-style-position**. Воно дозволяє задати всі властивості списку одночасно. З його допомогою стиль маркірованого списку з попереднього приклада міг би бути заданий так:

```
<UL style="list-style: square inside; width: 40%">
```

2.11.3.1. Лічильники

Для автоматичної нумерації в CSS використовуються *лічильники*. Лічильник — це ідентифікатор, якому присвоюються цілі значення за допомогою властивостей **counter-reset** і **counter-increment**. Поточне значення лічильника може використатися при генерації вмісту властивістю **content**.

За замовчуванням, лічильник позначається як *counter(name)* і його значенням є десяткові числа. Разом з тим ми можемо задавати стиль відображення значень лічильника у форматі *counter(name,style)*, де *style* може приймати будь-які значення властивості **list-style-type**. Відзначимо, що стиль *none* допустимо: *counter(name,none)* генерує порожній рядок. Приклади:

```
H1:before { content: counter(chapter, upper-latin) ". " }
H2:before { content: counter(section, upper-roman) " - " }
```

```
BLOCKQUOTE:after { content: " [" counter(bq, hebrew) "]" }
DIV.note:before { content: counter(notecnt, disc) " " }
P:before { content: counter(pcount, none) }
```

Елементи зі значенням *none* властивості **display** не змінюють значення лічильників. З іншого боку, елементи зі значенням *hidden* властивості **visibility** змінюють їхні значення, як звичайні елементи.

Підтримка: Не підтримуються

Не підтримуються

2.11.3.2. Початкове значення лічильника: властивість counter-reset

Синтаксис: counter-reset: [<ідентифікатор> <целое> ?]+ | none | [inherit](#)

Начально: none

Застосовно: до всіх елементів

Наслідувано: немає

Відсотки: не використовуються

Пристрою: [все](#)

Підтримка: Не підтримується

Не підтримується

Властивість **counter-reset** задає *початкові значення лічильників*. Його значенням є список імен лічильників, за кожним з яких може впливати необов'язкове ціле число; якщо таке число зазначене, то воно стає значенням лічильника, якщо ні, те лічильнику привласнюється значення 0.

Якщо елемент *i* задає початкове значення лічильника, і використовує його значення у властивості **content**, те значення використовується *після* присвоювання.

Дана властивість треба правилам каскадності. Тому наступні декларації

```
H1 { counter-reset: chapter 1 }
H1 { counter-reset: section -1 }
```

зададуть тільки значення лічильника section. Для завдання початкових значень обох лічильників їх потрібно вказати в одному правилі:

```
H1 { counter-reset: chapter 1 section -1 }
```

2.11.3.3. Зміна значення лічильника: властивість counter-increment

Синтаксис: counter-increment: [<ідентифікатор> <целое> ?]+ | none | [inherit](#)

Начально: none

Застосовно: до всіх елементів

Наслідувано: немає

Відсотки: не використовуються

Пристрою: [все](#)

Підтримка: Не підтримується

Не підтримується

Властивість **counter-increment** *змінює значення лічильників*. Його значенням є список імен лічильників, за кожним з яких може впливати необов'язкове ціле число; якщо таке число зазначене, то воно додається до поточного значення лічильника, якщо ні, те значення лічильника збільшується на 1. Ціле число може бути й нулем, і від'ємним.

Якщо використано лічильник, якому не привласнене початкове значення, то воно приймається рівним нулю. Якщо елемент *i* скидає, і змінює значення лічильника, то воно спочатку скидається, а потім

змінюється. Якщо елемент і змінює значення лічильника, і використовує його значення у властивості **content**, те значення використовується *після* зміни значення.

Наступний приклад показує, як нумерувати глави текстом виду "Розділ 1. ", а розділи — текстом виду "1.1 " і т.д.

```
H1:before {
  counter-increment: chapter; /* Збільшити chapter на 1 */
  counter-reset: section;    /* Обнулить section */
  content: "Глава " counter(chapter) ". ";
}
H2:before {
  counter-increment: section; /* Збільшити section на 1 */
  content: counter(chapter) "." counter(section) " ";
}
```

2.11.4. Маркери

2.11.4.1. Створення маркерів

Більшість блокових елементів в CSS генерують один осяжний прямокутник. Однак, CSS містить два механізми, що змушують елемент генерувати два прямокутники: головний прямокутник для вмісту елемента й *прямокутник маркера*, що може розташовуватися як усередині, так і поза головним прямокутником. На відміну від генерируемого вмісту, прямокутник маркера не впливає на позицію головного прямокутника. Більше загальний із цих механізмів називаються *маркери* й описаний у цьому розділі. Другий, більше обмежений по можливостях, але частіше використовуваний, називаються *списки* й описаний у наступному розділі.

Маркери створюються присвоюванням значення *marker* властивості **display** у псевдоелементах **:before** й **:after**. Прямокутник маркера форматується як однорядковий прямокутник, розташований поза головним прямокутником елемента. Він створюється тільки в тому випадку, коли властивість **content** псевдоелемента дійсно генерує зміст. Прямокутник маркера має рамку й заповнювач, але не має границь.

Для псевдоелемента **:before** базова лінія тексту прямокутника маркера вертикально вирівнюється по базовій лінії тексту першого рядка вмісту головного прямокутника, а для псевдоелемента **:after** — по базовій лінії тексту останнього рядка. Якщо головний прямокутник не містить тексту, то вирівнювання виробляється відповідно по верхньому й нижньому зовнішньому краї головного прямокутника.

Висота прямокутника маркера задається властивістю **line-height**. Вертикальне вирівнювання маркера усередині його прямокутника задається властивістю **vertical-align**.

Якщо властивість **width** має значення *auto*, те ширина прямокутника маркера визначається його вмістом, в інших випадках — значенням **width**. Якщо при цьому ширина вмісту більше ширини прямокутника, то обрізка визначається значенням властивості **overflow**. Прямокутник маркера може накладатися на головний прямокутник. Якщо ширина вмісту менше ширини прямокутника, то горизонтальне вирівнювання вмісту задається властивістю **text-align**.

Властивість **marker-offset** задає зсув прямокутника маркера по горизонталі щодо головного прямокутника, точніше, відстань між найближчими краями їхніх рамок.

Якщо властивість **display** має значення *marker* для вмісту, сгенерованного елементом, у якого значення **display** дорівнює *list-item*, те прямокутник маркера, сгенерований для **:before**, заміщає звичайний маркер елемента списку.

Наприклад, що впливає HTML-документ

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
  <TITLE>Створення списку з маркерами</TITLE>
```



```

<STYLE type="text/css">
  LI:before {
    display: marker;
    counter-increment: mycounter;
    content: counter(mycounter, lower-roman) ". ";
  }
</STYLE>
</HEAD>
<BODY>
  <OL>
    <LI>Перший елемент списку</LI>
    <LI>Другий елемент списку</LI>
    <LI>Третій елемент списку</LI>
  </OL>
</BODY>
</HTML>

```

буде відображатися так:

- i. Перший елемент списку.
- ii. Другий елемент списку.
- iii. Третій елемент списку.

Підтримка: Не підтримуються

Не підтримуються

2.11.4.2. Позиція маркера: властивість **marker-offset**

Синтаксис: `marker-offset: <розмір> | auto | inherit`

Начально: `auto`

Застосовно: до елементів типу `display: marker`

Наслідувано: немає

Відсотки: не використовуються

Пристрою: [визуальные](#)

Підтримка: Не підтримується

Не підтримується

Властивість **marker-offset** задає зсув *прямокутника маркера* по горизонталі щодо головного прямокутника, точніше, відстань між найближчими краями їхніх рамок. Воно може бути задане явно (<розмір>) або залишено на розсуд оглядача (*auto*). Розмір може бути від'ємним. Приклад:

```

LI:before { marker-offset: 3em }

```




Загальні(глобальні) атрибути елементів HTML Global Attributes

Події HTML 5 - це спеціальні глобальні атрибути, використовувані в тегах для виклику оброблювачів подій, написаних на різних мовах сценаріїв таких, як javascript і викликуваних, коли на сторінці відбувається яка-небудь дія. Події дозволяють зробити вашу сторінку динамічною.

Наступні атрибути є загальними для і можуть бути визначені на всіх HTML-елементів (навіть ті, які не визначені в даній специфікації).Ці атрибути підрозділяються на:

- базові атрибути (**class**, **id**, **style** і **title**), які визначають загальні властивості елементів;
- атрибути, що локалізують (**dir** і **lang**), які вказують на властивість мови, на якій написаний вміст елемента;
- нові атрибути до елементів(**accesskey**, **contenteditable**, **hidden**, **spellcheck**, **tabindex** і **translate**);
- обробка подій, які визначають реакцію даного елемента на різні події.

HTML Глобальні атрибути

Атрибут		Опис
accesskey		Визначає гарячу клавішу для включення / фокусує елемент
class		Задає один або кілька імен класів для елемента (відноситься до класу в таблиці стилів)
contenteditable		Визначає, чи буде вміст елемента є редагованим
contextmenu		Визначає контекстне меню для елемента. Контекстне меню з'являється, коли користувач клацає правою кнопкою миші на елементі
data-*		Використовується для зберігання даних користувача приватних на сторінку або застосуваннях
dir		Визначає напрямок тексту в елементі
draggable		Визначає, чи можна елемент перетягувати
dropzone		Зазначає чи копіювати, переміщувати або пов'язувати між собою дані, при перетягуванні
hidden		Вказує, що елементу поки ще немає, або вже немає у представленні
id		Вказує унікальний ідентифікатор для елемента
lang		Визначає мову вмісту елемента
spellcheck		Вказує, чи буде перевірка орфографії та граматики
style		Визначає CSS стиль вбудований для елемента
tabindex		Визначає порядку розміщення елементу
title		Визначає додаткову інформацію про елемент
translate		Визначає, чи буде вміст елемента трансльований



= додані в HTML5.

Обробка подій

Синтаксис: ім'я_події = сценарій


HTML підтримує обробку багатьох подій, які можуть відбуватися на комп'ютері-клієнті. Для обробки події потрібно привласнити елементу відповідний атрибут, ім'я якого є ім'ям цієї події, а значенням - сценарій (або виклик функції, або набір операторів). Значення такого атрибута може містити спеціальні символи.

Наступний приклад використовують оператори мови JavaScript для обробки двох подій, пов'язаних із кнопкою "Submit", а саме виводить підказку в рядку стану, коли курсор миші попадає на кнопку, і очищає рядок стану, коли курсор миші виявляється за її межами. Зверніть увагу, що значення атрибутів укладені в одинарні лапки через те, що подвійні лапки використані усередині них.

```
<INPUT type=submit  
  onmouseover='window.status="Ви заповнили всю форму?";'  
  onmouseout='window.status="";'>
```

Якщо документ містить обробку подій, то мова сценаріїв, прийнята за замовчуванням, повинна бути задана у заголовку документа в метаописувачі





```
<META http-equiv="Content-Script-Type" content="text/javascript">
```


 = додані в HTML5.

HTML атрибути подій

Атрибути подій вікна






Події, викликувані на об'єкт вікна (застосовуються з тегом <body>):

Атрибут	Опис
onafterprint	 Скрипт запуститься після того, як документ буде роздрукований.
onbeforeprint	 Скрипт запуститься перш, ніж документ буде роздрукований.
onbeforeunload	 Скрипт запуститься до завантаження документа.
onblur	 Скрипт запуститься, коли вікно втратить фокус.
onerror	 Скрипт запуститься, якщо відбудеться помилка.
onfocus	 Скрипт запуститься, коли вікно одержить фокус.
onhaschange	 Скрипт запуститься, після зміни документа.
onload	Код події буде виконаний після повного завантаження сторінки, тобто коли весь уміст веб-сторінки повністю завантажений, включаючи зображення, зовнішні таблиці стилів і зовнішні скрипти.
onmessage	 Скрипт запуститься при спрацюванні повідомлення.
onoffline	 Скрипт запуститься, коли документ перейде в автономний режим.
ononline	 Скрипт запуститься, коли документ увійде в мережу.
onpagehide	 Скрипт запуститься, коли вікно буде приховано.
onpageshow	 Скрипт запуститься, коли вікно стане видимим.
onpopstate	 Скрипт запуститься при зміні вікна історії.
onredo	 Скрипт запуститься, коли документ почне виконувати повтор.
onresize	 Скрипт запуститься при зміні розмірів вікна.
onstorage	 Скрипт запуститься при завантаженні документа.

onundo		Скрипт запуститься при виконанні команди скасувати в документі.
onunload		Скрипт запуститься при виході користувача з документа.

Події форми

Події спрацьовують на дії усередині Html-Форми (можуть застосовуватися до всіх елементів HTML, але найчастіше застосовуються до елементів форми):

Атрибут		Опис
onblur		Скрипт запуститься, коли елемент втратить фокус.
onchange		Скрипт запуститься при зміні елемента.
oncontextmenu		Скрипт запуститься при виклику контекстного меню.
onfocus		Скрипт запуститься, коли елемент одержить фокус.
onformchange		Скрипт запуститься при зміні форми.
onforminput		Скрипт запуститься, коли користувач буде вводити дані у форму.
oninput		Скрипт запуститься, коли елемент стане одержувати користувацьке введення.
oninvalid		Скрипт запуститься, коли елемент стане недійсний.
onselect		Скрипт запуститься при виборі елемента.
onsubmit		Скрипт запуститься при відправленні форми.

Події клавіатури



Події, викликувані клавіатурою (можуть застосовуватися до всіх елементів HTML):

Атрибут	Опис
onkeydown	Скрипт запуститься, коли буде натиснута клавіша.
onkeypress	Скрипт запуститься після того, як клавіша була натиснута й відпущена.
onkeyup	Скрипт запуститься при відпусканні клавіші.

Події миші








Події, викликувані діями мишкою або аналогічними діями користувача (можуть застосовуватися до всіх елементів HTML):

Атрибут	Опис
onclick	Код події буде виконаний по кличу лівою кнопкою миші на елементі.
ondblclick	Скрипт запуститься після подвійного кличу миші.

ondrag		Скрипт запуститься при перетаскуванні елемента.
ondragend		Скрипт запуститься після операції перетаскування.
ondragenter		Скрипт запуститься, коли елемент буде перенесений на припустиму зону падіння.
ondragleave		Скрипт запуститься, коли елемент відпустять у діючій зоні падіння.
ondragover		Скрипт запуститься, якщо елемент тягнуть за припустиму зону падіння.
ondragstart		Скрипт запуститься на початку операції перетаскування елемента.
ondrop		Скрипт запуститься після перетаскування елемента при падінні.
onmousedown		Скрипт запуститься при натисканні кнопки миші.
onmousemove		Скрипт запуститься, коли покажчик миші почне переміщатися.
onmouseout		Скрипт запуститься, коли покажчик миші буде переміщатися з елемента.
onmouseover		Скрипт запуститься, коли покажчик миші буде переміщатися над елементом.
onmouseup		Скрипт запуститься при відпусканні кнопки миші.
onmousewheel		Скрипт запуститься при використанні колеса миші.
onscroll		Скрипт запуститься під час прокручування смуги прокручування елемента.

Події медіа-файлів

Події, викликувані для медіа файлів, таких як відео, зображення й аудіо (можуть застосовуватися до всіх елементів HTML, але найчастіше застосовуються до таких елементів, як `<audio>`, `<embed>`, ``, `<object>` і `<video>`):

Атрибут		Опис
onabort		Скрипт запуститься при перериванні.
oncanplay		Скрипт запуститься, коли файл буде готовий, щоб почати програвання (коли буде досить буферизирован).
oncanplaythrough		Скрипт запуститься, коли файл може бути відтворений повністю (від початку до кінця), не зупиняючись для буферизації.
ondurationchange		Скрипт запуститься при зміні довжини тривалості медіа файлу.
onemptied		Скрипт запуститься, коли відбулася якась неполадка й файл став недоступний (наприклад при розриві зв'язку з інтернетом).
onended		Скрипт запуститься, коли аудіо/відео файл дійде до кінця (наприклад для виводу повідомлення: спасибі за прослуховування/перегляд).
onerror		Скрипт запуститься, якщо виникне помилка при завантаженні файлу.
onloadeddata		Скрипт запуститься при завантаженні даних медіа файлу.

onloadedmetadata		Скрипт запуститься, коли завантажені позначка-дані (наприклад розмір і тривалість).
onloadstart		Скрипт запуститься при завантаженні файлу перш, ніж почне завантажуватися насправді.
onpause		Скрипт запуститься, якщо відтворення файлу припинене (нажали паузу) користувачем або програмно.
onplay		Скрипт запуститься, коли файл готовий почати відтворюватися.
onplaying		Скрипт запуститься при відтворенні файлу.
onprogress		Скрипт запускається, коли браузер перебуває в процесі одержання даних медіа файлу.
onratechange		Скрипт запускається, щораз, при зміні режиму відтворення (наприклад, коли користувач перемикається на повільне відтворення або режим швидкого перемотування).
onreadystatechange		Скрипт запускається, щораз, при зміні стану готовності (стан готовності відслідковує стан даних файлу).
onseeked		Скрипт запускається, коли атрибут <code>seeked</code> у тегів <code>audio</code> або <code>video</code> має значення "false", тобто поточна позиція відтворення знайдена.
onseeking		Скрипт запускається, коли атрибут <code>seeking</code> у тегів <code>audio</code> або <code>video</code> має значення "true", тобто поточна позиція відтворення переміщається.
onstalled		Скрипт запускається, коли браузер не може одержати дані медіа файлу з будь-якої причини.
onsuspend		Скрипт запускається, коли одержання даних медіа файлу припиняється, до закінчення повного завантаження з будь-якої причини.
ontimeupdate		Скрипт запускається, коли позиція відтворення змінилася (наприклад, коли користувач вибрав крапку відтворення на багато далі від поточної).
onvolumechange		Скрипт запускається щораз при зміні значення гучності звуку (включаючи повне відключення звуку).
onwaiting		Скрипт запускається, коли медіа файл зупинений, але очікується поновлення запуску (наприклад, коли файл робить паузу для буферизації даних, що залишилися).