





## Лабораторна робота №6

### Тема. Проектування *Web*-документів. Сторіка форм.

**Мета.** Створення *Web*-сторінки з формами для введення інформації та вивчення **HTML** тегів: `<form>`, `<input>`, `<button>`, `<textarea>`, `<fieldset>`, `<legend>`, `<select>`, `<option>`, `<optgroup>`, `<keygen>`, `<keygen>`, `<progress>`, `<meter>`, `<datalist>`

### Теги мови HTML (продовження).

#### Теги HTML. Частина 4(рівень 4).

Тег	Призначення
<code>&lt;form&gt;</code>	Визначає форму
<code>&lt;input&gt;</code>	Визначає поле введення
<code>&lt;textarea&gt;</code>	Визначає текстову область
<code>&lt;button&gt;</code>	Визначає кнопку
<code>&lt;select&gt;</code>	Визначає список з можливістю вибору
<code>&lt;optgroup&gt;</code>	Визначає групу варіантів
<code>&lt;option&gt;</code>	Визначає елемент у поле списку
<code>&lt;label&gt;</code>	Визначає мітку для елемента керування форми
<code>&lt;fieldset&gt;</code>	Визначає набір полів
<code>&lt;legend&gt;</code>	Визначає заголовок набору полів
<code>&lt;isindex&gt;</code>	Не рекомендується. Визначає поле введення з одного рядка
<code>&lt;keygen&gt;</code>	 Визначає область для виведення, переважно за допомогою скриптів.
<code>&lt;progress&gt;</code>	 Використовується для відображення прогресу завершеності завдання. Зміна значення відбувається через JavaScript.
<code>&lt;meter&gt;</code>	 Використовується для виведення значення в деякому відомому діапазоні. Використовується переважно для відображення числових значень, наприклад, кількості результатів пошуку, об'єму рідини, тиску та ін.
<code>&lt;datalist&gt;</code>	 Створює список варіантів, які можна вибирати при наборі в текстовому полі. Спочатку цей список прихований і стає доступним при отриманні полем фокусу або наборі тексту.

## Форми

### 1. Загальний опис

Всі розглянуті нами дотепер елементи забезпечували тільки пасивна взаємодія користувача з оглядачем — перегляд Веб-сторінок і перехід до інших Веб-сторінок за допомогою гіперпосилань. Однак, у багатьох випадках Веб має потребу в активній взаємодії з користувачем, тобто в одержанні від нього інформації. Для інтерактивної взаємодії з користувачем мова HTML містить т.зв. *форми*. Форма — це розділ HTML-документа, що поряд зі звичайним змістом містить *керуючі елементи* (поля введення, кнопки, прапорці, перемикачі, меню й т.д.), а також *мітки* цих елементів. Користувач заповнює форму, змінюючи її керуючі елементи (наприклад, шляхом введення тексту

або вибору пункту меню), а потім пересилає заповнену форму для обробки серверу; звичайно це Веб-сервер або сервер електронної пошти.

Таким чином, центральною частиною будь-якої форми є її керуючі елементи. Кожний з них має ім'я, задане атрибутом **name** і діюче в межах поточної форми. Крім того, він має початкове значення й поточні значення, які є символьними рядками. Початкове значення елемента задається його атрибутами й стає поточним значенням елемента при першому відображенні форми або при її повторній ініціалізації. Надалі користувач може змінити поточне значення елемента. Коли заповнена форма пересилається для обробки, пересилається й частина її керуючих елементів у вигляді пар ім'я/значення. Такі керуючі елементи називаються *успішними*.

HTML підтримує наступні типи керуючих елементів:

### Кнопки (buttons)

Існують три типи кнопок:

- *кнопки пересилання форми*: при натисканні пересилають форму для обробки серверу;
- *кнопки скидання форми*: при натисканні ініціалізують форму, скидаючи значення всіх її елементів у початкові;
- *кнопки загального виду*: при натисканні на таку кнопку відбувається виклик пов'язаного із цією подією сценарію.

Кнопки можуть бути створені за допомогою елементів **BUTTON** або **INPUT**.

### Прапорці (checkboxes)

Прапорець — це елемент, що має два стани: обраний або не обраний. Прапорець обраний, якщо встановлено його атрибут **checked**. При пересиланні форми успішними є тільки обрані прапорці. Прапорці створюються за допомогою елемента **INPUT**.

### Перемикачі (radio buttons)

Перемикачі подібні до прапорців, але з однією відмінністю: якщо кілька перемикачів мають те саме ім'я елемента, те обраний може бути тільки один з них. Іншими словами, вони реалізують набір взаємовиключних варіантів. Перемикачі створюються за допомогою елемента **INPUT**.

### Меню (menus)

Меню дозволяють користувачеві здійснювати вибір з декількох варіантів. Меню створюється елементом **SELECT** у сполученні з елементами **OPTGROUP** і **OPTION**.

### Поля введення тексту (text input)

Є два елементи, що забезпечують введення тексту користувачем. Елемент **INPUT** створює однорядкове, а елемент **TEXTAREA** — багаторядкове поле введення. В обох випадках уведений текст стає поточним значенням елемента.

### Селектори файлів (file select)

Цей керуючий елемент дозволяє користувачеві вибрати файл для передачі його вмісту серверу при пересиланні форми. Селектори файлів створюються за допомогою елемента **INPUT**.

### Сховані елементи (hidden controls)

Можливе створення керуючих елементів, які не відображаються оглядачем, але значення яких передаються при пересиланні форми. Вони забезпечують збереження необхідних параметрів між сеансами обміну інформацією між клієнтом і сервером, що інакше була б загублена. Для створення схованих елементів використовується елемент **INPUT**.

### Об'єктні елементи (object controls)

Форми можуть містити об'єкти загального виду, уміст яких передається при пересиланні форми. Такі елементи створюються за допомогою елемента **OBJECT**.

## 2. Створення форми: елемент FORM

**Синтаксис:** <FORM>...</FORM> (блоковий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
action=URI (URI оброблювача форми)  
method = get | post (метод HTTP пересилання форми)  
enctype = тип файлу (тип файлу для пересилання форми)  
accept = типи файлів (список припустимих типів файлів)  
accept-charset = кодування (список припустимих кодувань символів)  
name = CDATA (назва форми для сценаріїв)  
target = фрейм (фрейм для відображення результатів)  
onsubmit = сценарій (пересилання форми)  
onreset = сценарій (ініціалізація форми)

Елемент **FORM** використовується для створення *інтерактивних форм* і служить контейнером для керуючих елементів, що утворюють форму. Крім керуючих елементів він може містити сценарії й будь-які блокові елементи, крім **FORM**. Даний елемент може втримуватися в елементах **BLOCKQUOTE**, **BODY**, **BUTTON**, **DD**, **DEL**, **DIV**, **FIELDSET**, **FORM**, **IFRAME**, **INS**, **LI**, **MAP**, **NOFRAMES**, **NOSCRIPT**, **OBJECT**, **TD** і **TH**.

Обов'язковий атрибут **action** задає URI програми, що приймає й обробляє результати заповнення форми користувачем. Звичайно це CGI-сценарій або Java-програма, що працює на Веб-сервері. Крім того, у якості **action** можливе завдання URI електронної пошти виду **mailto:name@server.com**.

Спосіб пересилання форми визначається атрибутами **method** й **enctype**. Він полягає в наступному:

1. Формується набір даних форми як послідовність пара імені-елемента/поточн-значення для всіх успішних керуючих елементів.
2. Цей набір даних кодується відповідно до типу файлу, заданим атрибутом **enctype**.
3. Діодированні дані передаються серверу відповідно до методу HTTP, заданим атрибутом **method**:
  - Якщо **method=get** (прийнято за замовчуванням), то до URI оброблювача форми додається '?' і набір даних форми, закодований відповідно до типу файлу "application/x-www-form-urlencoded". Отриманий URI передається Веб-серверу в запиті HTTP "GET". У цьому випадку всі дані форми повинні бути символами базової латиниці, а їхній припустимий обсяг обмежений.
  - Если **method=post**, то оглядач виконує запиті HTTP "POST" з URI запиту, заданим в **method**, і тілом повідомлення, створеним відповідно до типу файлу, заданим в **enctype** (звичайно це "multipart/form-data").

Елемент **FORM** може також мати наступні необов'язкові атрибути:

- Атрибут **accept** задає список типів файлів, розділених комами, які підтримуються сервером оброблювача форми. Оглядач може використати цей список для отфільтовки тільки припустимих файлів у селекторах файлів (див. елемент **INPUT** з **type=file**), але сучасні оглядачі звичайно ігнорують цей атрибут.
- Атрибут **accept-charset** задає список кодувань символів, розділених комами або пробілами, які підтримуються сервером оброблювача форми. За замовчуванням містить рядок "UNKNOWN" і звичайно інтерпретується як кодування символів документа, що містить даний елемент **FORM**.
- Атрибут **name** містить назва форми, на яке можна посилатися в таблицях стилів і сценаріях. Цей атрибут зберігається тільки з метою сумісності зі старими оглядачами; у цей час рекомендується використати замість нього атрибут id.

- Атрибут **target** задає ім'я документа, у якому повинен відображатися результат пересилання форми.

Крім того, даний елемент підтримує обробку двох нестандартних подій:

- **onsubmit** — відбувається при пересиланні форми для обробки;
- **onreset** — відбувається при скиданні всіх значень форми.

Наступний приклад форми містить поля введення, перемикачі й кнопки скидання й пересилання форми:

```
<FORM action="http://mysite.com/cgi-bin/adduser" method="post">
  <P>
    <LABEL for="firstname">Ім'я: </LABEL>
    <INPUT type="text" id="firstname"><BR>
    <LABEL for="lastname">Прізвище: </LABEL>
    <INPUT type="text" id="lastname"><BR>
    <LABEL for="email">E-mail: </LABEL>
    <INPUT type="text" id="email"><BR>
    <INPUT type="radio" name="sex" value="Male" checked="checked">
      Чоловік<BR>
    <INPUT type="radio" name="sex" value="Female"> Жінка<BR>
    <INPUT type="submit" value="Send"> <INPUT type="reset">
  </P>
</FORM>
```

### 3. Введення даних: елемент INPUT

**Синтаксис:** <INPUT> (умісту й кінцевого тегу немає)

**Атрибути:** `id, class, style, title, lang, dir, події`  
`type = text | password | checkbox | radio | submit | reset |`  
`file | hidden | image | button` (тип введення)  
`name = CDATA` (ім'я керуючого елемента)  
`value = CDATA` (початкове значення елемента)  
`checked = checked` (стан обраний для прапорців і перемикачів)  
`readonly = readonly` (тільки читання для тексту й паролів)  
`disabled = disabled` (елемент заборонений)  
`size = CDATA` (початкова ширина елемента)  
`maxlength = число` (максимальна кількість символів тексту)  
`src = URI` (URI графічного образу кнопки)  
`alt = текст` (альтернативний текст для кнопки)  
`usemap = URI` (URI клієнтської карти посилань)  
`ismap = ismap` (серверна карта посилань)  
`accesskey = символ` (ім'я клавіші швидкого доступу)  
`tabindex = число` (номер у порядку натискання табуляції)  
`accept = типи файлів` (список припустимих типів файлів)  
`onfocus = сценарій` (при одержанні елементом фокуса)  
`onblur = сценарій` (при втраті елементом фокуса)  
`onselect = сценарій` (при виборі тексту елемента)  
`onchange = сценарій` (при зміні значення елемента)

Елемент **INPUT** створює керуючий елемент форми, призначений для *уведення даних користувачем*. Він може втримуватися в будь-яких блокових або текстових елементах, крім **BUTTON**, але багато оглядачів відображають його тільки тоді, коли він перебуває усередині елемента **FORM**. Даний елемент використовується для створення різних керуючих елементів, залежно від значення атрибута **type**:

#### text

Однорядковий текст введення (прийнято за замовчуванням). Атрибут **value** задає початкове значення тексту, **size** — розмір поля введення в символах, а **maxlength** — максимально можлива кількість символів у даному полі. Атрибут **readonly** забороняє зміна тексту поля.

#### password

Поле введення пароля. Єдина відмінність від поля введення тексту полягає в тому, що вводять символи, що, маскуються, звичайно заміною на зірочки. Однак, уведена інформація передається серверу як звичайний текст, тому цей

елемент не забезпечує рівня безпеки, достатнього для передачі номерів кредитних карт або інший істотно важливої інформації.

#### checkbox

Прапорець(ознака) Атрибут **value** задає значення цього елемента, коли він обраний, а атрибут **checked** — чи обраний спочатку.

#### radio

Перемикач. Атрибут **value** задає значення цього елемента, коли він обраний, а атрибут **checked** — чи обраний спочатку.

#### reset

Кнопка скидання форми. Необов'язковий атрибут **value** перекриває текст кнопки, прийнятий за замовчуванням.

#### submit

Кнопка пересилання форми. Необов'язковий атрибут **value** перекриває текст кнопки, прийнятий за замовчуванням. Якщо форма містить кілька таких кнопок, то атрибут **name** використовується для передачі імені кнопки оброблювачеві форми, що дозволяє останньому здійснювати різні дії залежно від того, яка із кнопок пересилання форми була натиснута.

#### image

Графічна кнопка пересилання форми. Атрибут **src** задає URI графічного образу кнопки, а **alt** — альтернативний текст напису на кнопці. При щиглику мишею по такій кнопці відбувається пересилання форми, причому координати щиглика миші передаються серверу у вигляді *name.x=x-value* й *name.y=y-value*, де *name* — ім'я кнопки, а *x-value* й *y-value* — координати в пікселях щодо верхнього лівого кута образу. Атрибут **usemap** указує на клієнтську карту посилань і його значенням повинне бути ім'я закладки, задане атрибутом **name** відповідного елемента **MAP**. Атрибут **ismap** указує на серверну карту посилань.

#### button

Кнопка загального виду. Атрибут **value** задає текст кнопки, а атрибут **onclick** повинен задавати сценарій, викликуваний при натисканні цієї кнопки.

#### file

Селектор файлів. Атрибут **value** задає початкове ім'я файлу, але оглядачі звичайно ігнорують його з міркувань безпеки. Необов'язковий атрибут **accept** задає список типів файлів, розділених комами, які підтримуються сервером оброблювача форми. Цей список може використатися для отфільтровки тільки припустимих файлів, але сучасні оглядачі звичайно ігнорують цей атрибут. Форма, що містить селектор файлів, повинна мати атрибути **method=post** й **enctype="multipart/form-data"**.

#### hidden

Скритий елемент. Атрибут **disabled** забороняє даний керуючий елемент. Це означає, що елемент не може одержати фокус, не доступний шляхом натискання табуляції й не може бути успішним. Спосіб відображення забороненого елемента залежить від оглядача.

Необов'язкові атрибути **accesskey** й **tabindex** застосовні до всіх типів елементів, крім схованих:

- Атрибут **accesskey** задає символ Unicode як клавіша швидкого доступу до елемента.
- Значенням атрибута **tabindex** повинне бути ціле число в діапазоні від 0 до 32767. Це число задає номер даного елемента 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 елементах першим одержує фокус той, котрий раніше зустрічається в документі.

Даний елемент підтримує обробку наступних нестандартних подій:

- **onfocus** — відбувається, коли елемент одержує фокус;
- **onblur** — відбувається, коли елемент губить фокус;
- **onselect** — відбувається, коли обраний текст у поле типу text або password;
- **onchange** — відбувається, коли елемент губить фокус і його значення змінилося з моменту попереднього одержання фокуса.

Приклади використання елемента **INPUT** наведені в попередньому розділі.

#### 4. Кнопки: елемент **BUTTON**

**Синтаксис:** <BUTTON>...</BUTTON> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
type = submit|reset|button (тип уведення)  
name = CDATA (ім'я керуючого елемента)  
value = CDATA (початкове значення кнопки)  
disabled = disabled (елемент заборонений)  
accesskey = символ (ім'я клавіші швидкого доступу)  
tabindex = число (номер у порядку натискання табуляції)  
onfocus = сценарій (при одержанні елементом фокуса)  
onblur = сценарій (при втраті елементом фокуса)

Елемент **BUTTON** створює *кнопку* форми. Він може містити будь-які текстові елементи, крім **A**, **BUTTON**, **IFRAME**, **INPUT**, **LABEL**, **SELECT** і **TEXTAREA**, і будь-які блокові елементи, крім **FORM** і **FIELDSET**. Він може втримуватися в будь-яких текстових і блокових елементах, крім **BUTTON**.

Кнопки, створені цим елементом, подібні до кнопок, створеним елементом **INPUT**, але вони можуть мати вміст. Атрибут **name** задає ім'я даного керуючого елемента, а **value** — його початкове значення. Атрибут **type** задає тип кнопки:

##### **submit**

Кнопка пересылки форми (прийнято за замовчуванням).

##### **reset**

Кнопка сброса формы.

##### **button**

Кнопка общего вида.

Атрибут **disabled** *забороняє* даний керуючий елемент. Це означає, що елемент не може одержати фокус, не доступний шляхом натискання табуляції й не може бути успішним. Спосіб відображення забороненого елемента залежить від оглядача. Інші необов'язкові атрибути:

- Атрибут **accesskey** задає символ Unicode як клавіша швидкого доступу до елемента.
- Значенням атрибута **tabindex** повинне бути ціле число в діапазоні від 0 до 32767. Це число задає номер даного елемента 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 елементах першим одержує фокус той, котрий раніше зустрічається в документі.

Даний елемент підтримує обробку наступних нестандартних подій:

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

Наступний приклад є переробкою з використанням елемента **BUTTON** приклада форми, наведеного вище:

```
<FORM action="http://mysite.com/cgi-bin/adduser" method="post">
  <P>
    <LABEL for="firstname">Ім'я: </LABEL>
    <INPUT type="text" id="firstname"><BR>
    <LABEL for="lastname">Прізвище: </LABEL>
    <INPUT type="text" id="lastname"><BR>
    <LABEL for="email">E-mail: </LABEL>
```



```

        <INPUT type="text" id="email"><BR>
        <INPUT type="radio" name="sex" value="Male" checked="checked">
Чоловік<BR>
        <INPUT type="radio" name="sex" value="Female"> Жінка<BR>
        <BUTTON name="submit" value="submit" type="submit">Послати</BUTTON>
        <BUTTON name="reset" type="reset">Скинути</BUTTON>
    </P>
</FORM>

```

## 5. Меню: елементи SELECT, OPTGROUP й OPTION

**Синтаксис:** <SELECT>...</SELECT> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
 name = CDATA (ім'я керуючого елемента)  
 size = число (кількість видимих опцій)  
 multiple = *multiple* (дозволяє множинний вибір)  
 disabled = *disabled* (елемент заборонений)  
 tabindex = число (номер у порядку натискання табуляції)  
 onfocus = сценарій (при одержанні елементом фокуса)  
 onblur = сценарій (при втраті елементом фокуса)  
 onchange = сценарій (при зміні значення елемента)

**Підтримка:** Відповідає стандарту (5.0+). Атрибути **disabled** й **tabindex** ігноруються (4.0+)

Елемент **SELECT** створює керуючий елемент форми, призначений для вибору користувачем опцій зі списку. Звичайно цей елемент відображається у вигляді меню, що випадає. Його вмістом можуть бути тільки елементи **OPTGROUP** і **OPTION**, які визначають відповідний список опцій. Він може втримуватися в будь-яких блокових і текстових елементах, крім **BUTTON**.

Атрибут **name** задає ім'я даного керуючого елемента. Атрибут **size** указує оглядачам розмір видимої частини списку опцій. Якщо загальна кількість опцій більше, те оглядач повинен забезпечити прокручування вікна списку, щоб забезпечити доступ до всіх опцій.

За замовчуванням, користувач може вибрати тільки одну опцію зі списку. Щоб дозволити йому вибір декількох опцій одночасно, варто включити в даний елемент атрибут **multiple**. У цьому випадку кожна з обраних опцій передається при пересиланні форми як окрема пара ім'я/значення.

Атрибут **disabled** забороняє даний керуючий елемент. Це означає, що елемент не може одержати фокус, не доступний шляхом натискання табуляції й не може бути успішним. Спосіб відображення забороненого елемента залежить від оглядача.

Значенням атрибута **tabindex** повинне бути ціле число в діапазоні від 0 до 32767. Це число задає номер даного елемента 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 елементах першим одержує фокус той, котрий раніше зустрічається в документі.

Даний елемент підтримує обробку наступних нестандартних подій:

- **onfocus** — відбувається, коли елемент одержує фокус;
- **onblur** — відбувається, коли елемент губить фокус;
- **onchange** — відбувається, коли елемент губить фокус і його значення змінилося з моменту попереднього одержання фокуса.

**Синтаксис:** <OPTGROUP>...</OPTGROUP> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
 label = текст (мітка групи опцій)  
 disabled = *disabled* (елемент заборонений)

**Синтаксис:** <OPTION>...</OPTION> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
 value = CDATA (ім'я опції)  
 label = текст (мітка опції)

selected = selected (опція начально обрана)  
disabled = disabled (елемент заборонений)  
**Підтримка:** Атрибути disabled й label ігноруються (5.0+)  
Атрибути disabled й label ігноруються (4.0+)

Елементи **OPTION** й **OPTGROUP** призначені для завдання відповідно *опцій* і *груп опцій*. Вони можуть уживатися тільки усередині елемента **SELECT**. Елемент **OPTGROUP** може містити тільки елементи **OPTION**, а елемент **OPTION** — тільки звичайний текст (включаючи спеціальні символи).

Обов'язковий атрибут **label** елемента **OPTGROUP** задає мітку групи опцій, що може виводитися оглядачем при відображенні списку опцій. Сучасні оглядачі не підтримують цей елемент, але його використання усередині елемента **SELECT** дозволяє зробити логічну структуру меню більше наочною, хоча й не впливає поки на його відображення.

Тепер розглянемо елемент **OPTION**. Атрибут **value** задає ім'я цього елемента, що використовується при пересиланні форми; якщо цього атрибута ні, те ім'ям опції вважається вміст даного елемента. Атрибут **selected** указує, що дана опція є обраною в момент відображення меню. Меню може містити тільки одну опцію із цим атрибутом, якщо в нього немає атрибута **multiple**. Атрибут **label** задає текст опції, що відображається оглядачем; якщо цього атрибута ні, те відображається вміст даного елемента. Атрибут **disabled** уживається так само, як в елементі **SELECT**.

Приклад меню, що містить всі три описаних елементи:

```
<FORM action="http://mysite.com/cgi-bin/selector" method="post">
  <P>Який оглядач ви використаєте?
  <SELECT name="browser">
    <OPTGROUP label="Netscape Navigator">
      <OPTION label="4.x або вище">Netscape Navigator 4.x або
вище</OPTION>
      <OPTION label="3.x">Netscape Navigator 3.x</OPTION>
      <OPTION label="2.x">Netscape Navigator 2.x</OPTION>
      <OPTION label="1.x">Netscape Navigator 1.x</OPTION>
    </OPTGROUP>
    <OPTGROUP label="Microsoft Internet Explorer">
      <OPTION label="4.x або вище">Internet Explorer 4.x або
вище</OPTION>
      <OPTION label="3.x">Internet Explorer 3.x</OPTION>
      <OPTION label="2.x">Internet Explorer 2.x</OPTION>
      <OPTION label="1.x">Internet Explorer 1.x</OPTION>
    </OPTGROUP>
    <OPTGROUP label="Opera">
      <OPTION label="3.x або вище">Opera 3.x або вище</OPTION>
      <OPTION label="2.x">Opera 2.x</OPTION>
    </OPTGROUP>
    <OPTION selected="selected">Інші</OPTION>
  </SELECT>
  <INPUT type="submit" value="Send"> <INPUT type="reset">
</P>
</FORM>
```

## 6. Многострочное текстовое поле: элемент TEXTAREA

**Синтаксис:** <TEXTAREA>...</TEXTAREA> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
name = CDATA (ім'я керуючого елемента)  
**rows** = число (кількість видимих рядків)  
**cols** = число (кількість видимих стовпців)  
readonly = readonly (тільки читання)  
disabled = disabled (елемент заборонений)  
accesskey = символ (ім'я клавіші швидкого доступу)  
tabindex = число (номер у порядку натискання табуляції)  
onfocus = сценарій (при одержанні елементом фокуса)  
onblur = сценарій (при втраті елементом фокуса)  
onselect = сценарій (при виборі тексту елемента)  
onchange = сценарій (при зміні значення елемента)



Елемент **TEXTAREA** створює керуючий елемент форми, призначений для *многострочного введення даних користувачем*. Його вмістом повинен бути звичайний текст, включаючи спеціальні символи. Він може втримуватися в будь-яких блокових і текстових елементах, крім **BUTTON**. Уміст елемента **TEXTAREA** задає початкове значення відображуваного тексту. При пересиланні форми атрибут **name** задає ім'я даного керуючого елемента, а поточний уміст поля введення — його значення.

Обов'язкові атрибути **rows** й **cols** задають, відповідно, кількість рядків і стовпців вікна оглядача, що займає при відображенні даний елемент. Ці атрибути є скоріше рекомендацією оглядачеві, чим строгим обмеженням; користувач може ввести більше тексту, чим дозволяють зазначені атрибути, тому оглядачі забезпечують прокручування поля введення по горизонталі й вертикалі. Теоретично оглядачі можуть приймати введення тексту будь-якого розміру, але на практиці цей розмір звичайно обмежений обсягом в 32Кб або 64Кб. Необов'язковий атрибут **readonly** забороняє зміну первісного вмісту поля.

Атрибут **disabled** забороняє даний керуючий елемент. Це означає, що елемент не може одержати фокус, не доступний шляхом натискання табуляції й не може бути успішним. Спосіб відображення забороненого елемента залежить від оглядача. Інші необов'язкові атрибути:

- Атрибут **accesskey** задає символ Unicode як клавіша швидкого доступу до елемента.
- Значенням атрибута **tabindex** повинне бути ціле число в діапазоні від 0 до 32767. Це число задає номер даного елемента 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 елементах першим одержує фокус той, котрий раніше зустрічається в документі.

Даний елемент підтримує обробку наступних нестандартних подій:

- **onfocus** — відбувається, коли елемент одержує фокус;
- **onblur** — відбувається, коли елемент губить фокус.
- **onselect** — відбувається, коли обраний текст у поле введення;
- **onchange** — відбувається, коли елемент губить фокус і його значення змінилося з моменту попереднього одержання фокуса.

Приклад використання елемента **TEXTAREA** наведений у п. 1.8.8.

## 7. Мітки: елемент LABEL

**Синтаксис:** <LABEL>...</LABEL> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події  
for = IDREF (ідентифікатор керуючого елемента)  
accesskey = символ (ім'я клавіші швидкого доступу)  
onfocus = сценарій (при одержанні елементом фокуса)  
onblur = сценарій (при втраті елементом фокуса)

Елемент **LABEL** створює текстову *мітку* для керуючого елемента форми. Його вмістом можуть бути будь-які текстові елементи, крім **LABEL**. Він може втримуватися в будь-яких блокових і текстових елементах, крім **BUTTON**.

Деякі керуючі елементи, наприклад, кнопки пересилання форми, містять явні мітки, які задаються їхнім атрибутом **label**. Елемент **LABEL** призначений для присвоювання міток тим керуючим елементам, які цього атрибута не мають. Кожен елемент **LABEL** асоціюється рівно з одним керуючим елементом форми. Це може бути зроблено двома способами:

- Ми можемо вказати в атрибуті **for** елемента **LABEL** ідентифікатор відповідного керуючого елемента. У цьому випадку значення атрибута **for** повинне збігатися зі значенням атрибута **id** керуючого елемента, наприклад:

```
<FORM action="..." method="post">
<TABLE>
  <TR>
```

```

        <TD><LABEL for="fname">Ім'я</LABEL>
        <TD><INPUT type="text" name="firstname" id="fname">
    <TR>
        <TD><LABEL for="lname">Прізвище</LABEL>
        <TD><INPUT type="text" name="lastname" id="lname">
</TABLE>
</FORM>

```

- Якщо елемент **LABEL** не має атрибута **for**, то асоційований з ним керуючий елемент повинен бути його вмістом, наприклад:

```

<FORM action="..." method="post">
<P>
    <LABEL>Ім'я
        <INPUT type="text" name="firstname">
    </LABEL>
    <LABEL>Прізвище
        <INPUT type="text" name="lastname">
    </LABEL>
</P>
</FORM>

```

Необов'язковий атрибут **accesskey** задає символ Unicode як клавіша швидкого доступу до елемента.

Даний елемент підтримує обробку наступних нестандартних подій:

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

## 8. Групи керуючих елементів: елементи **FIELDSET** й **LEGEND**

**Синтаксис:** <FIELDSET>...</FIELDSET> (блоковий елемент)

**Атрибути:** id, class, style, title, lang, dir, події

**Синтаксис:** <LEGEND>...</LEGEND> (текстовий елемент)

**Атрибути:** id, class, style, title, lang, dir, події

**accesskey** = символ (ім'я клавіші швидкого доступу)

Елемент **FIELDSET** визначає *групу керуючих елементів* форми. Поєднуючи взаємозалежні керуючі елементи в групи, автори можуть розбити форму на кілька частин, полегшуючи користувачеві її заповнення. Умістом цього елемента є елемент **LEGEND**, що задає титул групи елементів, за яким можуть впливати будь-які блокові й/або текстові елементи, включаючи інший елемент **FIELDSET**. Він може втримуватися в елементах **BLOCKQUOTE**, **BODY**, **DD**, **DEL**, **DIV**, **FIELDSET**, **FORM**, **IFRAME**, **INS**, **LI**, **MAP**, **NOFRAMES**, **NOSCRIPT**, **OBJECT**, **TD** і **TH**.

Елемент **LEGEND** задає титул групи елементів. Він може містити будь-які текстові елементи й може вживатися тільки на початку елемента **FIELDSET**. Необов'язковий атрибут **accesskey** задає символ Unicode як клавіша швидкого доступу до елемента.

Приклад форми, розбитої на три групи керуючих елементів:

```

<FORM action="..." method="post">
<P>
<FIELDSET>
    <LEGEND>Особиста інформація</LEGEND>
    Прізвище: <INPUT name="personal_lastname" type="text" tabindex="1">
    Ім'я: <INPUT name="personal_firstname" type="text" tabindex="2">
    Адреса: <INPUT name="personal_address" type="text" tabindex="3">
</FIELDSET>
<FIELDSET>
    <LEGEND>Історія хвороби</LEGEND>
    <INPUT name="history_illness"
        type="checkbox"
        value="Smallpox" tabindex="20"> Віспа
    <INPUT name="history_illness"
        type="checkbox"
        value="Mumps" tabindex="21"> Свинка
    <INPUT name="history_illness"
        type="checkbox"
        value="Dizziness" tabindex="22"> Запаморочення

```

```

<INPUT name="history_illness"
      type="checkbox"
      value="Sneezing" tabindex="23"> Кашель
</FIELDSET>
<FIELDSET>
<LEGEND>Поточне лікування</LEGEND>
Чи приймаєте ви зараз які-небудь ліки?
<INPUT name="medication_now"
      type="radio"
      value="Yes" tabindex="35">Так
<INPUT name="medication_now"
      type="radio"
      value="No" tabindex="35">Немає

Якщо ви приймаєте зараз ліки, перелічіть їх нижче:
<TEXTAREA name="current_medication"
          rows="20" cols="50"
          tabindex="40">
</TEXTAREA>
</FIELDSET>
</P>
</FORM>

```

## 9. Форми в HTML5

У HTML5 в елементів форм, з'явилися додаткові атрибути і їх значення. Завдання нових форм HTML5 полягає в тому щоб полегшувати створення скриптів або навіть замінити собою типові функції, які раніше здійсненні були лише за допомогою скриптів.

### Нові теги

Тег `<keygen>`

Теги `<keygen>` `</keygen>` використовуються для шифровки й розшифрування даних.

Тег `<output>`

Теги `<output>` `</output>` створюють область для виводу роботи скрипта.

### Тег `<FORM>`

Нові атрибути і їх значення в тегу `form`:

1. **`autocomplete="on/off"`** - включає/відключає автозаповнення у форм, якщо в браузері автозаповнення відключене, то атрибут не працює,
2. **`novalidate="novalidate"`** - відключає вбудовану перевірку даних, уведених користувачем у поля `<input type="email" />` і `<input type="url" />`

### Тег `<INPUT >`

Нові атрибути і їх значення в тегу `input`:

1. **`autocomplete="on/off"`** - включає/відключає автозаповнення в тегу `input`, якщо в браузері автозаповнення відключене, то атрибут не працює,
2. **`autofocus="autofocus"`** - установка фокуса на елемент,
3. **`form="idname"`** - зв'язує тег `input`, з формою по атрибуту **`id="idname"`**, за умови що тег `input` перебуває поза формою,
4. **`formaction="URL"`** - указує на адресу програми яка буде обробляти дані введені в тег `input`, аналогічний атрибуту **`action=" "`** у тегу **`form`**,
5. **`formenctype=" "`** - указує на спосіб кодування даних при їхнього відправлення на сервер, аналогічний атрибуту `enctype=" "`, у тегу `form`, має наступні значення:
6. **`application/x-www-form-urlencoded`** - не латинські символи кодуються їх шістнадцятковими кодами, пробіли замінюються знаком `+`, є значенням за замовчуванням, **`multipart/form-data`** - дані не кодуються, **`text/plain`** - дані не кодуються, але пробіли замінюються знаком `+`,
7. **`formmethod="get/post"`** - указує яким методом відправити дані, `post` або `get`,

8. **formnovalidate="formnovalidate"** - відключає вбудовану перевірку даних, аналогічний атрибуту **novalidate=""** у тегу **form**
9. **formtarget="ім'я\_вікна/\_blank/\_self/\_parent/\_top"** - вказує на вікно в яке буде завантажуватися, результат обробки даних, аналогічний атрибуту **target=""** у тегу **form** **list="idname"** - вказує на ім'я ідентифікатора **id="idname"** тегу **datalist**, у якому перебуває список варіантів, що з'являються при введенні тексту в поле,
10. **max=""** - вказує на максимальне число, яке можна ввести в поле,
11. **min=""** - вказує на мінімальне число, яке можна ввести в поле,
12. **multiple="multiple"** - дозволяє вказати декілька **e-mail** адрес, у поле через кому, або завантажити кілька файлів,
13. **pattern=""** - вказує на регулярне вираз, виходячи з якого приймається рішення про вірність або не вірності введених даних, атрибут призначений для створення власних полів введення,
14. **placeholder=""** - вказує на рядок, який знаходиться в текстовому полі, вона зникає коли користувач починає вводити дані,
15. **readonly="readonly"** - вказує на текстове поле, у яке користувач не може ввести дані,
16. **required="required"** - вказує на текстове поле, яке обов'язково повинне бути заповненим, якщо користувач спробує відправити дані не заповнивши це поле, то браузер видасть про це повідомлення вигляді спливаючої підказки,
17. **step=""** - вказує на крок, при введенні чисел.

#### Атрибут type тега <INPUT>

У мові HTML5 в атрибуті type="" , додалося наступні значення:

1. **type="color"** - створює віджет вибору кольору,
2. **type="date"** - створює поле для введення дати, за допомогою віджета календаря (далі ВК),
3. **type="datetime"** - створює поле для введення дати й часу (ВК),
4. **type="datetime-local"** - створює поле для введення дати й місцевого часу (ВК),
5. **type="email"** - створює поле для введення e-mail,
6. **type="month"** - створює поле для введення місяця (ВК),
7. **type="number"** - створює поле для введення числа,
8. **type="range"** - створює повзунок,
9. **type="search"** - створює пошукове поле,
10. **type="tel"** - створює поле для введення телефонного номера,
11. **type="time"** - створює поле для введення дати й часу (ВК), без часового пояса,
12. **type="url"** - створює поле для введення url,
13. **type="week"** - створює поле для введення тижня (ВК).

### Завдання до лабораторної роботи №6

Розробити Web – сторінку введення інформації за допомогою форм (без відправки на Web-сервер: **<form action="#">** ) для Web –проекту. На Web – сторінці студент виводить придуману інформацію про товари фірми(підприємство) згідно варіанту за допомогою форм. При натисненні кнопки відправки( **<input type="submit" value="Відправити інформацію"/>**) здійснювати можливий контроль введення.

При розробці сторінки студент може використовувати будь-які теги мови HTML, але для кожного студента, згідно варіанту, задається набір обов'язкових тегів та їх атрибутів, що повинні бути присутні на Web – сторінці.

Приклад сторінки.

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<meta name="Author" lang="ua" content="Гаштемеленко Яваскриптик Перлович">
<meta http-equiv="Content-Script-Type" content="text/javascript">
<meta name="Description" content="Приклад Web - сторінки, Web Developer's Handbook. HTML Reference.">
<meta name="Keywords" content="Web - сторінка, новини, компанія ">
<title>Приклад. Форми.
</title>
<script language="JavaScript">
function fnd1()
{
document.frmfnd.txta1.value="";
var result;
    str=document.frmfnd.txtf1.value;
    mask=new RegExp("0|1|2|3|4|5|6|7|8|9","g");
result=str.match(mask);
document.frmfnd.txta1.value=(result.length);
}
function fnd2()
{
document.frmfnd.txta2.value="";
var result;
    str=document.frmfnd.txtf2.value;
    mask=new RegExp("k[a-z]*o","g");
result=str.match(mask);
document.frmfnd.txta2.value=(result);
}
</script>
<style>
body {
    font-family: 'Palatino Linotype', serif;
    max-width: 600px;
    padding: 0px 30px;
}
h1 {
    margin-bottom: 0px;
}
p {
    margin-top: 0px;
}

fieldset {
    margin-bottom: 15px;
    padding: 10px;
}

legend {
    padding: 0px 3px;
    font-weight: bold;
    font-variant: small-caps;
}

label {
    width: 150px;
    display: inline-block;
    vertical-align: top;
    margin: 6px;
}

em {
    font-weight: bold;
    font-style: normal;

```

```

    color: #f00;
}

input, textarea {
    width: 249px;
}

textarea {
    height: 100px;
}

select {
    width: 254px;
}

input[type=checkbox] {
    width: 10px;
}

input[type=submit] {
    width: 170px;
    padding: 10px;
}

input:focus {
    background: #eaeaea;
}
</style>
</head>

<body bgcolor=#00ffcc>

<form action="#">
<p><i>Заповніть форму. Обов'язкові поля помічені </i><em>*</em></p>
<fieldset>
    <legend>Контактна інформація</legend>
    <label for="name">Ім'я <em>*</em></label>
    <input id="name" placeholder="Петро Степанович" autofocus required><br>
    <label for="telephone">Телефон</label>
    <input type="tel" id="telephone" placeholder="+38 0xx xxx xxxx" ><br>
    <label for="email">Email <em>*</em></label>
    <input id="email" placeholder="ps_t@ukr.net" pattern="\S+@[a-z]+\.[a-z]+" required><br>
</fieldset>
<fieldset>
    <legend>Персональна інформація</legend>
    <label for="age">Вік<em>*</em></label>
    <input type="number" min="18" max="120" step="1" value=21 placeholder=21><br>
    <label for="gender">Стать</label>
    <select id="gender">
        <option value="male">Чоловік</option>
        <option value="female">Жінка</option>
    </select><br>
    <label for="zp">Заробітня плата</label>
    <input id="zp" type="range" min="1000" max="12000" step="10" value=5000><br>
    <label for="comments">Перерахуйте особисті якості </label>
    <textarea id="comments" placeholder="спокійний, щедрий, ввічливий, добрий, ....."></textarea>
    <br>
    <label for="date_n">Дата народження<em>*</em></label>
    <input id="date_n" type="date" min="1920-01-01" max="2010-01-01"><br>
    <br>
    <label for="addr_n">Адреса вашої Веб-сторінки</label>
    <input id="addr_n" type="url" placeholder="http://hjp.cs.chnu.cv.ua" ><br>

```



```

</fieldset>

<fieldset>
  <legend>Виберіть ваші любимі тварини </legend>
  <label for="zebra"><input id="zebra" type="checkbox"> Зебра</label>
  <label for="cat"><input id="cat" type="checkbox"> Кішак</label>
  <label for="anaconda"><input id="anaconda" type="checkbox"> Анаконда</label>
  <label for="human"><input id="human" type="checkbox"> Чоловік</label>
  <label for="elephant"><input id="elephant" type="checkbox"> Слон</label>
  <label for="wildebeest"><input id="wildebeest" type="checkbox"> Антилопа</label>
  <label for="pigeon"><input id="pigeon" type="checkbox"> Голуб</label>
  <label for="crab"><input id="crab" type="checkbox"> Краб</label>
</fieldset>
<fieldset>
<datalist id="dl_fruits">
  <select id="favoriteFruite">
    <option value="Яблоко" label="apple">
    <option value="Ананас" label="anas">
    <option value="Абрикос" label="apricot">
    <option value="Кавун" label="watermelon">
    <option value="Авокадо" label="avocado">
    <option value="Апельсин" label="Orange">
  </select>
</datalist>
<legend>Виберіть Ваш </legend>
<label for="mycolor">любимий колір</label>
<input id="mycolor" type="color" value="#00ff00"><br>
  <label for="fruit">фрукт</label> <input id="fruit" list="dl_fruits" placeholder="Кавун" >
</fieldset>
Заповнення анкети: <progress min=100 max="200"> Анкета заповняється...</progress>
<br>
<p><input type="submit" value="Відправити інформацію"></p>
<fieldset>
  <legend>Віджети </legend>
  <br>
  <label for="date_tm">Дата datetime </label>
  <input id="date_tm" name="date_tm" type="datetime" ><br>
  <label for="date_tml">Дата time local </label>
  <input id="date_tml" name="date_tml" type="datetime-local" ><br>
  <label for="date_month">Дата month </label>
  <input id="date_month" name="date_month" type="month" ><br>
  <label for="date_time">Дата time </label>
  <input id="date_time" name="date_time" type="time" ><br>
  <label for="date_week">Дата week </label>
  <input id="date_week" name="date_week" type="week" ><br>
</fieldset>
</form>

Ваш багаж важить: <meter min="5" max="70" value="28">28 кілограм</meter><br>
Ваш багаж важить: <meter min="5" max="220" high="10" >...кілограм</meter><br>
Ваш багаж важить: <meter min="5" max="100" high="70" value="79">79 кілограм</meter>*
<p><small>* За багаж вагою більше 70 кг. треба доплатити.</small></p>

Виконання плану <meter value="0.75">75%</meter> <br>

Температура води :
<meter max="100" low="10" high="60" value="5"> Низька </meter>
<meter max="100" low="10" high="60" value="30"> Нормальна </meter>
  <meter max="100" low="10" high="60" value="40"> Нормальна </meter>
    <meter max="100" low="10" high="60" value="50"> Нормальна </meter>
  <meter max="100" low="10" high="60" value="80"> Гаряча </meter>
  <meter max="100" low="10" high="60" value="100"> Окріп </meter>
<br>

```

```

<form id="frm_img" name="frm_img" method="get" action="">
  <fieldset>
    <legend>Карті посилань </legend>
    <br>
    
    <map name="shapes">
      <area shape=rect coords="50,50,100,100"> <!-- the hole in the red box -->
      <area shape=rect coords="25,25,125,125" href="red.html" alt="Red box.">
      <area shape=circle coords="200,75,50" href="green.html" alt="Green circle.">
      <area shape=poly coords="325,25,262,125,388,125" href="blue.html" alt="Blue triangle.">
      <area shape=poly coords="450,25,435,60,400,75,435,90,450,125,465,90,500,75,465,60"
        href="yellow.html" alt="Yellow star.">
    </map>

  </fieldset>
</form>

<h1 align="center">Пошук</h1>
<form id="frmfnd" name="frmfnd" method="post" action="">
  <p align="left">Введіть текст:</p>
  <input name="txtf1" type="text" size="85%"/>
  <p align="left">Результат: кількість цифр</p>
  <p>
    <input name="txta1" type="text" size="85%"/>
  </p>
  <p align="left">
    <input name="btn1" type="button" onclick="javascript:fnd1()" value="Пошук" />
  </p>
  <hr>
  <input type="submit" value="Очистити форму">
  <hr color='yellow'>
  <p align="left">Введіть текст:</p>
  <input name="txtf2" type="text" size="85%"/>
  <p align="left">Результат: слова що починаються на k і закінчуються на o</p>
  <p>
    <input name="txta2" type="text" size="85%"/>
  </p>
  <p align="left">
    <input name="btn2" type="button" onclick="javascript:fnd2()" value="Пошук" />
  </p>
</form>
</body>
</html>

```

## Варіанти завдань

**Етап 5.** Розробка сторінки введення інформації. Вивчення тегів: <form>, <input>, <button> <textarea> <fieldset>, <legend>, <select>, <option>, <optgroup>.

## Варіанти завдань

Варіант	обов'язкові теги та їх атрибути
1	тер : <form> атрибути - action, accept{-}, name, dir, accept-charset, method; тер : <input> атрибути - maxlength, max {+}, src, formtarget {+}, formmethod {+}, disabled, value, dir, min {+}, autofocus {+}, max {+}, minlength {?}, checked; тер : <input> атрибути - type=' tel', type=' date ', type=' month ', type=' number ', type=' radio', type=' datetime ', type=' search', type=' tel'; тер : <textarea> атрибути - autofocus {+}, style, disabled, class, maxlength {+}, dir, readonly; тер : <button> атрибути - formaction {+}, formtarget {+}, autofocus {+}, name; тер : <legend> атрибути - title, class, dir; тер : <isindex>{-} атрибути - dir, style, id; тер : <label> атрибути - class, form, style; тер : <optgroup> атрибути - label, dir, disabled; тер : <output> атрибути - lang, title, id, for; тер : <summary> атрибути - dir,

	lang, title; тер : <meter> атрибути - min, id, low, style, dir; тер : <progress> атрибути - class, style, value;
2	тер : <form> атрибути - name, class, lang, novalidate{+}, enctype, action; тер : <input> атрибути - name, dir, minlength{?}, formmethod{+}, formenctype{+}, type, src, min{+}, size, required{+}, width{+}, height{+}, multiple{+}; тер : <input> атрибути - type='range', type='week', type='url', type='tel', type='checkbox', type='text', type='date', type='search'; тер : <textarea> атрибути - form{+}, lang, wrap{+}, readonly, required{+}, name, autofocus{+}; тер : <button> атрибути - formtarget{+}, value, form{+}, formaction{+}; тер : <isindex>{-} атрибути - action, dir, lang; тер : <fieldset> атрибути - class, lang, id; тер : <select> атрибути - autofocus{+}, disabled, title, multiple, name; тер : <legend> атрибути - id, dir, style; тер : <details> атрибути - title, style, class; тер : <output> атрибути - id, class, form, for; тер : <meter> атрибути - title, min, class, id, value; тер : <progress> атрибути - style, id, dir;
3	тер : <form> атрибути - dir, target, accept{-}, accept-charset, id, action; тер : <input> атрибути - alt, formenctype{+}, lang, maxlength, title, pattern{+}, width{+}, src, max{+}, formtarget{+}, class, minlength{?}, dirname; тер : <input> атрибути - type='url', type='week', type='image', type='datetime-local', type='reset', type='radio', type='password', type='month'; тер : <textarea> атрибути - placeholder{+}, minlength{+-}, cols, dirname, name, rows, lang; тер : <button> атрибути - formmethod{+}, value, autofocus{+}, type; тер : <optgroup> атрибути - disabled, id, label; тер : <label> атрибути - lang, title, class; тер : <legend> атрибути - lang, id, dir; тер : <isindex>{-} атрибути - dir, title, style; тер : <progress> атрибути - id, lang, class; тер : <summary> атрибути - lang, dir, style; тер : <keygen> атрибути - form, keytype, id, title, autofocus; тер : <meter> атрибути - style, low, dir, high, title;
4	тер : <form> атрибути - lang, style, accept-charset, target, autocomplete{+}, enctype; тер : <input> атрибути - type, height{+}, step{+}, autofocus{+}, multiple{+}, title, autocomplete{+}, alt, formtarget{+}, formenctype{+}, formmethod{+}, required{+}, maxlength; тер : <input> атрибути - type='month', type='week', type='email', type='tel', type='checkbox', type='image', type='file', type='url'; тер : <textarea> атрибути - name, wrap{+}, form{+}, title, placeholder{+}, cols, disabled; тер : <button> атрибути - autofocus{+}, value, formmethod{+}, type; тер : <optgroup> атрибути - title, disabled, lang; тер : <isindex>{-} атрибути - id, class, prompt; тер : <legend> атрибути - id, style, dir; тер : <label> атрибути - class, form, title; тер : <keygen> атрибути - style, dir, keytype, challenge, lang; тер : <datalist> атрибути - dir, style, title; тер : <meter> атрибути - style, lang, class, max, low; тер : <details> атрибути - dir, open, id;
5	тер : <form> атрибути - autocomplete{+}, action, class, dir, style, lang; тер : <input> атрибути - style, maxlength, autofocus{+}, step{+}, minlength{?}, autocomplete{+}, checked, alt, min{+}, formtarget{+}, multiple{+}, value, maxlength; тер : <input> атрибути - type='submit', type='checkbox', type='week', type='month', type='email', type='search', type='url', type='tel'; тер : <textarea> атрибути - disabled, dirname, lang, title, required{+}, class, style; тер : <button> атрибути - form{+}, type, name, disabled; тер : <legend> атрибути - lang, title, dir; тер : <isindex>{-} атрибути - class, id, lang; тер : <optgroup> атрибути - style, class, id; тер : <label> атрибути - id, form, title; тер : <summary> атрибути - title, class, id; тер : <progress> атрибути - max, id, lang; тер : <meter> атрибути - class, title, max, min, high; тер : <keygen> атрибути - id, dir, form, style, lang;
6	тер : <form> атрибути - target, title, lang, id, method, dir; тер : <input> атрибути - pattern{+}, formenctype{+}, dir, autofocus{+}, accept, style, size, formaction{+}, height{+}, min{+}, class, required{+}, max{+}; тер : <input> атрибути - type='tel', type='range', type='file', type='date', type='number', type='image', type='checkbox', type='month'; тер : <textarea> атрибути - title, form{+}, maxlength{+}, dir, autofocus{+}, required{+}, name; тер : <button> атрибути - name, form{+}, formtarget{+}, value; тер : <fieldset> атрибути - dir, class, id; тер : <label> атрибути - title, style, class; тер : <legend> атрибути - lang, id, style; тер : <optgroup> атрибути - id, dir, title; тер : <meter> атрибути - min, lang, id, title, high; тер : <keygen> атрибути - class, style, autofocus, dir, challenge; тер : <datalist> атрибути - title, dir, lang; тер : <progress> атрибути - class, max, id;
7	тер : <form> атрибути - method, action, novalidate{+}, enctype, style, class; тер : <input> атрибути - pattern{+}, value, min{+}, size, formaction{+}, form{+}, height{+}, max{+}, type, required{+}, style, minlength{?}, height{+}; тер : <input> атрибути - type='url', type='radio', type='datetime-local', type='submit', type='button', type='email', type='date', type='search'; тер : <textarea> атрибути - style, id, lang, title, rows, disabled,

	form {+}; тер : <button> атрибути - name, disabled, value, formenctype {+}; тер : <select> атрибути - lang, dir, style, multiple, class; тер : <legend> атрибути - title, id, class; тер : <label> атрибути - style, form, id; тер : <optgroup> атрибути - title, disabled, style; тер : <progress> атрибути - lang, style, class; тер : <details> атрибути - title, lang, id; тер : <datalist> атрибути - title, style, lang; тер : <output> атрибути - lang, title, form, for;
8	тер : <form> атрибути - novalidate{+}, enctype, accept{-}, style, accept-charset, method; тер : <input> атрибути - type, formtarget {+}, formaction {+}, alt, multiple {+}, maxlength, dir, title, height {+}, min {+}, formtarget {+}, max {+}, max {+}; тер : <input> атрибути - type='tel', type='image', type='password', type='radio', type='hidden', type='reset', type='range', type='button'; тер : <textarea> атрибути - autofocus {+}, placeholder {+}, readonly, id, disabled, wrap {+}, title; тер : <button> атрибути - disabled, formmethod {+}, form {+}, value; тер : <isindex>{-} атрибути - prompt, action, id; тер : <select> атрибути - id, name, required {+}, form {+}, autofocus {+}; тер : <option> атрибути - lang, id, selected, disabled; тер : <fieldset> атрибути - style, dir, id; тер : <summary> атрибути - dir, title, style; тер : <datalist> атрибути - id, class, title; тер : <progress> атрибути - lang, title, max; тер : <meter> атрибути - high, value, max, min, id;
9	тер : <form> атрибути - dir, id, enctype, novalidate{+}, action, accept{-}; тер : <input> атрибути - formmethod {+}, required {+}, alt, title, formaction {+}, step {+}, formtarget {+}, pattern {+}, max {+}, id, height {+}, height {+}, formenctype {+}; тер : <input> атрибути - type='month', type='tel', type='color', type='image', type='search', type='password', type='datetime-local', type='reset'; тер : <textarea> атрибути - id, placeholder {+}, wrap {+}, required {+}, name, title, disabled; тер : <button> атрибути - autofocus {+}, formenctype {+}, name, formtarget {+}; тер : <legend> атрибути - class, id, title; тер : <isindex>{-} атрибути - style, action, lang; тер : <label> атрибути - id, form, class; тер : <optgroup> атрибути - id, lang, title; тер : <details> атрибути - style, title, open; тер : <output> атрибути - title, lang, class, for; тер : <datalist> атрибути - id, dir, lang; тер : <keygen> атрибути - disabled, lang, style, title, name;
10	тер : <form> атрибути - method, dir, accept{-}, enctype, accept-charset, id; тер : <input> атрибути - size, max {+}, checked, formtarget {+}, form {+}, minlength {?}, height {+}, lang, formaction {+}, name, value, max {+}, formtarget {+}; тер : <input> атрибути - type='text', type='image', type='radio', type='button', type='datetime-local', type='file', type='password', type='number'; тер : <textarea> атрибути - name, minlength {+-}, form {+}, style, placeholder {+}, rows, dirname; тер : <button> атрибути - value, formmethod {+}, autofocus {+}, formnovalidate {+}; тер : <legend> атрибути - lang, title, dir; тер : <label> атрибути - title, style, form; тер : <option> атрибути - id, dir, title, label; тер : <select> атрибути - multiple, autofocus {+}, style, size, id; тер : <keygen> атрибути - disabled, class, keytype, title, style; тер : <details> атрибути - dir, lang, id; тер : <output> атрибути - class, id, style, for; тер : <datalist> атрибути - id, title, lang;
11	тер : <form> атрибути - lang, style, autocomplete{+}, dir, method, title; тер : <input> атрибути - maxlength, src, title, pattern {+}, type, alt, class, maxlength, formnovalidate {+}, formenctype {+}, size, formaction {+}, accept; тер : <input> атрибути - type='text', type='search', type='image', type='tel', type='number', type='radio', type='range', type='week'; тер : <textarea> атрибути - rows, lang, autofocus {+}, placeholder {+}, dirname, minlength {+-}, required {+}; тер : <button> атрибути - formenctype {+}, disabled, formmethod {+}, formaction {+}; тер : <label> атрибути - style, form, lang; тер : <legend> атрибути - class, lang, dir; тер : <option> атрибути - id, label, selected, class; тер : <select> атрибути - name, style, lang, id, required {+}; тер : <datalist> атрибути - dir, title, class; тер : <keygen> атрибути - autofocus, style, dir, keytype, form; тер : <summary> атрибути - dir, id, title; тер : <meter> атрибути - min, title, optimum, class, value;
12	тер : <form> атрибути - accept{-}, id, novalidate{+}, lang, method, accept-charset; тер : <input> атрибути - title, alt, maxlength, height {+}, formtarget {+}, multiple {+}, required {+}, class, size, formtarget {+}, max {+}, lang, form {+}; тер : <input> атрибути - type='submit', type='tel', type='number', type='email', type='week', type='password', type='datetime-local', type='datetime'; тер : <textarea> атрибути - cols, id, wrap {+}, name, class, disabled, lang; тер : <button> атрибути - formtarget {+}, value, formmethod {+}, autofocus {+}; тер : <select> атрибути - title, class, form {+}, autofocus {+}, dir; тер : <isindex>{-} атрибути - id, style, dir; тер : <label> атрибути - class, lang, style; тер : <fieldset> атрибути - class, style, title; тер : <output> атрибути - style, for, dir, class; тер : <datalist> атрибути - title, lang, style; тер : <details> атрибути - id, open, class; тер : <meter> атрибути - id, low, dir, style, lang;

13	<p>тер : &lt;form&gt; атрибути - accept{-}, enctype, action, class, title, novalidate{+}; тер : &lt;input&gt; атрибути - title, formtarget {+}, formmethod {+}, height {+}, formenctype {+}, size, autofocus {+}, formnovalidate {+}, required {+}, lang, multiple {+}, maxlength, max {+}; тер : &lt;input&gt; атрибути - type='checkbox', type='color', type='tel', type='date', type='email', type='month', type='datetime', type='number'; тер : &lt;textarea&gt; атрибути - autofocus {+}, form {+}, title, lang, rows, required {+}, maxlength {+}; тер : &lt;button&gt; атрибути - form {+}, formenctype {+}, formnovalidate {+}, formaction {+}; тер : &lt;select&gt; атрибути - title, dir, class, form {+}, name; тер : &lt;fieldset&gt; атрибути - dir, class, title; тер : &lt;label&gt; атрибути - lang, id, dir; тер : &lt;option&gt; атрибути - dir, lang, disabled, class; тер : &lt;details&gt; атрибути - style, dir, lang; тер : &lt;progress&gt; атрибути - class, title, dir; тер : &lt;datalist&gt; атрибути - title, style, id; тер : &lt;keygen&gt; атрибути - style, form, challenge, lang, id;</p>
14	<p>тер : &lt;form&gt; атрибути - id, accept-charset, style, class, name, action; тер : &lt;input&gt; атрибути - max {+}, formnovalidate {+}, height {+}, style, src, checked, lang, accept, title, dir, id, max {+}, formtarget {+}; тер : &lt;input&gt; атрибути - type='url', type='datetime', type='tel', type='file', type='number', type='submit', type='month', type='reset'; тер : &lt;textarea&gt; атрибути - cols, name, id, form {+}, rows, autofocus {+}, dirname; тер : &lt;button&gt; атрибути - formmethod {+}, formenctype {+}, value, formtarget {+}; тер : &lt;select&gt; атрибути - required {+}, disabled, dir, class, multiple; тер : &lt;legend&gt; атрибути - style, title, id; тер : &lt;optgroup&gt; атрибути - label, id, style; тер : &lt;label&gt; атрибути - lang, dir, title; тер : &lt;output&gt; атрибути - dir, style, lang, for; тер : &lt;meter&gt; атрибути - optimum, low, max, style, class; тер : &lt;datalist&gt; атрибути - lang, dir, title; тер : &lt;keygen&gt; атрибути - id, disabled, name, title, keytype;</p>
15	<p>тер : &lt;form&gt; атрибути - id, target, name, autocomplete{+}, accept-charset, dir; тер : &lt;input&gt; атрибути - dir, id, name, width {+}, src, multiple {+}, autofocus {+}, autocomplete {+}, formmethod {+}, step {+}, required {+}, class, max {+}; тер : &lt;input&gt; атрибути - type='file', type='submit', type='tel', type='hidden', type='tel', type='range', type='text', type='image'; тер : &lt;textarea&gt; атрибути - id, title, required {+}, lang, style, minlength {+}, name; тер : &lt;button&gt; атрибути - formtarget {+}, name, autofocus {+}, formnovalidate {+}; тер : &lt;label&gt; атрибути - class, dir, lang; тер : &lt;fieldset&gt; атрибути - style, class, dir; тер : &lt;option&gt; атрибути - id, dir, lang, label; тер : &lt;select&gt; атрибути - class, lang, title, autofocus {+}, required {+}; тер : &lt;meter&gt; атрибути - optimum, class, dir, low, max; тер : &lt;details&gt; атрибути - lang, open, title; тер : &lt;output&gt; атрибути - for, style, form, lang; тер : &lt;keygen&gt; атрибути - class, title, autofocus, keytype, form;</p>
16	<p>тер : &lt;form&gt; атрибути - lang, method, class, novalidate{+}, name, enctype; тер : &lt;input&gt; атрибути - width {+}, alt, formtarget {+}, autofocus {+}, autocomplete {+}, multiple {+}, dirname, formtarget {+}, required {+}, pattern {+}, src, formaction {+}, size; тер : &lt;input&gt; атрибути - type='email', type='date', type='url', type='number', type='button', type='image', type='datetime-local', type='submit'; тер : &lt;textarea&gt; атрибути - disabled, rows, autofocus {+}, minlength {+}, form {+}, maxlength {+}, id; тер : &lt;button&gt; атрибути - disabled, autofocus {+}, type, name; тер : &lt;label&gt; атрибути - class, for, lang; тер : &lt;optgroup&gt; атрибути - lang, class, label; тер : &lt;legend&gt; атрибути - lang, style, id; тер : &lt;select&gt; атрибути - form {+}, class, name, autofocus {+}, lang; тер : &lt;progress&gt; атрибути - dir, value, title; тер : &lt;datalist&gt; атрибути - dir, style, class; тер : &lt;meter&gt; атрибути - high, optimum, dir, title, class; тер : &lt;details&gt; атрибути - lang, open, style;</p>
17	<p>тер : &lt;form&gt; атрибути - accept{-}, name, action, method, id, novalidate{+}; тер : &lt;input&gt; атрибути - maxlength, max {+}, type, height {+}, disabled, name, max {+}, formenctype {+}, dirname, formnovalidate {+}, width {+}, autocomplete {+}, checked; тер : &lt;input&gt; атрибути - type='datetime', type='radio', type='datetime-local', type='color', type='button', type='url', type='month', type='submit'; тер : &lt;textarea&gt; атрибути - dirname, class, name, placeholder {+}, required {+}, readonly, minlength {+}; тер : &lt;button&gt; атрибути - name, formenctype {+}, autofocus {+}, form {+}; тер : &lt;fieldset&gt; атрибути - lang, style, title; тер : &lt;option&gt; атрибути - dir, disabled, selected, label; тер : &lt;label&gt; атрибути - id, class, for; тер : &lt;legend&gt; атрибути - title, lang, style; тер : &lt;output&gt; атрибути - class, form, lang, name; тер : &lt;keygen&gt; атрибути - disabled, name, title, challenge, autofocus; тер : &lt;progress&gt; атрибути - value, dir, max; тер : &lt;details&gt; атрибути - lang, class, open;</p>
18	<p>тер : &lt;form&gt; атрибути - method, lang, action, class, style, novalidate{+}; тер : &lt;input&gt; атрибути - class, step {+}, minlength {?}, lang, height {+}, formtarget {+}, dirname, max {+}, formaction {+}, maxlength, disabled, value, formenctype {+}; тер : &lt;input&gt; атрибути - type='button', type='email', type='datetime-local', type='range', type='number', type='tel', type='date', type='color'; тер : &lt;textarea&gt; атрибути - required {+}, rows, name, form {+}, dir, cols, maxlength {+}; тер : &lt;button&gt; атрибути - formenctype {+}, disabled, autofocus {+}, formaction {+}; тер : &lt;label&gt; атрибути - id, style, dir; тер : &lt;optgroup&gt;</p>

	атрибути - style, lang, class; тер : <isindex>{-} атрибути - id, prompt, dir; тер : <option> атрибути - selected, lang, disabled, value; тер : <keygen> атрибути - challenge, keytype, dir, autofocus, style; тер : <details> атрибути - title, open, lang; тер : <meter> атрибути - style, optimum, min, id, class; тер : <output> атрибути - name, style, class, title;
19	тер : <form> атрибути - target, method, accept-charset, accept{-}, enctype, id; тер : <input> атрибути - height {+}, size, id, multiple {+}, name, autofocus {+}, formtarget {+}, formaction {+}, height {+}, dirname, required {+}, formmethod {+}, formtarget {+}; тер : <input> атрибути - type='tel', type='file', type='color', type='week', type='submit', type='reset', type='datetime', type='password'; тер : <textarea> атрибути - form {+}, maxlength {+}, autofocus {+}, disabled, dirname, cols, rows; тер : <button> атрибути - form {+}, disabled, formmethod {+}, formnovalidate {+}; тер : <fieldset> атрибути - id, lang, class; тер : <option> атрибути - selected, dir, value, label; тер : <optgroup> атрибути - disabled, lang, dir; тер : <legend> атрибути - id, title, style; тер : <summary> атрибути - lang, id, class; тер : <datalist> атрибути - id, title, dir; тер : <output> атрибути - lang, dir, name, class; тер : <progress> атрибути - value, id, class;
20	тер : <form> атрибути - method, dir, title, target, name, id; тер : <input> атрибути - formnovalidate {+}, formmethod {+}, accept, height {+}, style, pattern {+}, autocomplete {+}, form {+}, lang, id, src, required {+}, multiple {+}; тер : <input> атрибути - type='reset', type='month', type='tel', type='color', type='button', type='search', type='password', type='image'; тер : <textarea> атрибути - lang, required {+}, readonly, disabled, dir, title, maxlength {+}; тер : <button> атрибути - formmethod {+}, formtarget {+}, autofocus {+}, formnovalidate {+}; тер : <option> атрибути - title, disabled, class, value; тер : <optgroup> атрибути - label, id, lang; тер : <label> атрибути - title, class, form; тер : <isindex>{-} атрибути - prompt, lang, class; тер : <output> атрибути - class, id, name, dir; тер : <summary> атрибути - style, dir, id; тер : <keygen> атрибути - class, keytype, name, id, disabled; тер : <progress> атрибути - title, dir, style;
21	тер : <form> атрибути - novalidate{+}, autocomplete{+}, accept{-}, name, class, dir; тер : <input> атрибути - src, disabled, title, autofocus {+}, formtarget {+}, multiple {+}, style, type, formenctype {+}, autocomplete {+}, checked, class, minlength {?}; тер : <input> атрибути - type='submit', type='checkbox', type='month', type='datetime-local', type='file', type='date', type='number', type='image'; тер : <textarea> атрибути - form {+}, minlength {+}, readonly, rows, wrap {+}, lang, cols; тер : <button> атрибути - formmethod {+}, type, value, autofocus {+}; тер : <label> атрибути - form, class, for; тер : <optgroup> атрибути - title, id, lang; тер : <legend> атрибути - class, title, dir; тер : <isindex>{-} атрибути - action, prompt, style; тер : <meter> атрибути - class, optimum, min, title, high; тер : <keygen> атрибути - title, disabled, name, style, autofocus; тер : <output> атрибути - style, lang, dir, name; тер : <details> атрибути - dir, class, open;
22	тер : <form> атрибути - accept-charset, dir, autocomplete{+}, target, lang, title; тер : <input> атрибути - multiple {+}, formtarget {+}, disabled, value, formmethod {+}, lang, checked, size, required {+}, style, autofocus {+}, name, step {+}; тер : <input> атрибути - type='datetime', type='month', type='submit', type='url', type='button', type='color', type='file', type='text'; тер : <textarea> атрибути - minlength {+}, class, autofocus {+}, required {+}, dirname, title, name; тер : <button> атрибути - form {+}, autofocus {+}, disabled, type; тер : <option> атрибути - value, selected, title, class; тер : <select> атрибути - title, size, required {+}, id, disabled; тер : <label> атрибути - for, form, class; тер : <optgroup> атрибути - label, style, disabled; тер : <meter> атрибути - lang, id, high, class, style; тер : <progress> атрибути - id, value, max; тер : <output> атрибути - for, title, lang, id; тер : <datalist> атрибути - title, style, dir;
23	тер : <form> атрибути - title, method, style, class, dir, id; тер : <input> атрибути - maxlength, lang, value, max {+}, src, max {+}, size, name, form {+}, min {+}, pattern {+}, formtarget {+}, autocomplete {+}; тер : <input> атрибути - type='range', type='radio', type='datetime', type='hidden', type='submit', type='image', type='text', type='button'; тер : <textarea> атрибути - style, cols, disabled, dir, name, maxlength {+}, rows; тер : <button> атрибути - type, formaction {+}, formenctype {+}, autofocus {+}; тер : <isindex>{-} атрибути - id, action, prompt; тер : <label> атрибути - dir, id, lang; тер : <option> атрибути - class, selected, style, disabled; тер : <optgroup> атрибути - lang, title, dir; тер : <progress> атрибути - style, dir, id; тер : <meter> атрибути - dir, lang, class, low, max; тер : <datalist> атрибути - dir, id, lang; тер : <summary> атрибути - title, class, style;



24	<p>ter : &lt;form&gt; атрибути - accept{-}, name, dir, enctype, class, target; ter : &lt;input&gt; атрибути - formaction {+}, max {+}, step {+}, autofocus {+}, pattern {+}, autocomplete {+}, height {+}, value, src, disabled, formtarget {+}, minlength {?}, dir; ter : &lt;input&gt; атрибути - type=' color', type=' submit', type=' search', type=' password', type=' image', type=' datetime ', type=' week', type=' date '; ter : &lt;textarea&gt; атрибути - dir, minlength {+-}, style, placeholder {+}, wrap {+}, maxlength {+}, dirname; ter : &lt;button&gt; атрибути - formaction {+}, formenctype {+}, formnovalidate {+}, type; ter : &lt;label&gt; атрибути - id, form, class; ter : &lt;option&gt; атрибути - disabled, selected, label, style; ter : &lt;isindex&gt;{-} атрибути - lang, id, style; ter : &lt;optgroup&gt; атрибути - label, style, title; ter : &lt;summary&gt; атрибути - lang, class, dir; ter : &lt;meter&gt; атрибути - min, low, title, lang, value; ter : &lt;progress&gt; атрибути - class, id, dir; ter : &lt;details&gt; атрибути - lang, dir, open;</p>
25	<p>ter : &lt;form&gt; атрибути - autocomplete{+}, lang, class, style, enctype, accept-charset; ter : &lt;input&gt; атрибути - checked, autofocus {+}, alt, accept, dirname, required {+}, formnovalidate {+}, formaction {+}, lang, height {+}, width {+}, multiple {+}, max {+}; ter : &lt;input&gt; атрибути - type=' button', type=' checkbox', type=' radio', type=' image', type=' tel', type=' text', type=' reset', type=' tel'; ter : &lt;textarea&gt; атрибути - name, autofocus {+}, wrap {+}, style, disabled, placeholder {+}, cols; ter : &lt;button&gt; атрибути - name, value, form {+}, formenctype {+}; ter : &lt;isindex&gt;{-} атрибути - prompt, dir, title; ter : &lt;fieldset&gt; атрибути - title, style, class; ter : &lt;option&gt; атрибути - style, disabled, label, id; ter : &lt;select&gt; атрибути - multiple, id, required {+}, name, size; ter : &lt;datalist&gt; атрибути - style, class, lang; ter : &lt;meter&gt; атрибути - dir, high, id, class, style; ter : &lt;summary&gt; атрибути - title, dir, class; ter : &lt;progress&gt; атрибути - value, max, id;</p>
26	<p>ter : &lt;form&gt; атрибути - class, dir, action, title, novalidate{+}, enctype; ter : &lt;input&gt; атрибути - height {+}, type, pattern {+}, checked, autocomplete {+}, height {+}, maxlength, formnovalidate {+}, formtarget {+}, max {+}, required {+}, class, lang; ter : &lt;input&gt; атрибути - type=' number ', type=' tel', type=' week', type=' url', type=' range ', type=' password', type=' radio', type=' file'; ter : &lt;textarea&gt; атрибути - wrap {+}, title, dir, cols, name, class, rows; ter : &lt;button&gt; атрибути - autofocus {+}, name, formnovalidate {+}, type; ter : &lt;fieldset&gt; атрибути - class, dir, lang; ter : &lt;optgroup&gt; атрибути - dir, id, class; ter : &lt;isindex&gt;{-} атрибути - id, prompt, title; ter : &lt;option&gt; атрибути - lang, disabled, id, selected; ter : &lt;meter&gt; атрибути - high, class, dir, title, max; ter : &lt;progress&gt; атрибути - lang, value, id; ter : &lt;keygen&gt; атрибути - lang, form, dir, keytype, id; ter : &lt;datalist&gt; атрибути - lang, style, title;</p>
27	<p>ter : &lt;form&gt; атрибути - id, accept-charset, lang, title, enctype, accept{-}; ter : &lt;input&gt; атрибути - maxlength, formnovalidate {+}, multiple {+}, name, size, autofocus {+}, height {+}, formtarget {+}, value, pattern {+}, max {+}, formenctype {+}, width {+}; ter : &lt;input&gt; атрибути - type=' url', type=' image', type=' color', type=' tel', type=' radio', type=' search', type=' email ', type=' datetime '; ter : &lt;textarea&gt; атрибути - readonly, required {+}, dir, id, form {+}, disabled, minlength {+-}; ter : &lt;button&gt; атрибути - value, formmethod {+}, formaction {+}, type; ter : &lt;label&gt; атрибути - id, lang, class; ter : &lt;legend&gt; атрибути - dir, class, lang; ter : &lt;fieldset&gt; атрибути - style, dir, class; ter : &lt;optgroup&gt; атрибути - lang, title, id; ter : &lt;summary&gt; атрибути - class, id, lang; ter : &lt;datalist&gt; атрибути - id, style, dir; ter : &lt;progress&gt; атрибути - value, id, style; ter : &lt;details&gt; атрибути - class, dir, title;</p>
28	<p>ter : &lt;form&gt; атрибути - title, enctype, target, action, name, novalidate{+}; ter : &lt;input&gt; атрибути - style, src, lang, formmethod {+}, maxlength, formenctype {+}, name, size, autofocus {+}, formtarget {+}, formtarget {+}, pattern {+}, step {+}; ter : &lt;input&gt; атрибути - type=' submit', type=' tel', type=' datetime-local ', type=' range ', type=' hidden', type=' checkbox', type=' datetime ', type=' tel'; ter : &lt;textarea&gt; атрибути - minlength {+-}, id, autofocus {+}, readonly, placeholder {+}, disabled, wrap {+}; ter : &lt;button&gt; атрибути - formnovalidate {+}, form {+}, formmethod {+}, autofocus {+}; ter : &lt;label&gt; атрибути - title, dir, form; ter : &lt;optgroup&gt; атрибути - label, title, dir; ter : &lt;fieldset&gt; атрибути - title, id, lang; ter : &lt;legend&gt; атрибути - title, class, id; ter : &lt;details&gt; атрибути - id, open, dir; ter : &lt;datalist&gt; атрибути - dir, id, style; ter : &lt;summary&gt; атрибути - dir, lang, title; ter : &lt;keygen&gt; атрибути - form, disabled, autofocus, keytype, dir;</p>
29	<p>ter : &lt;form&gt; атрибути - target, style, action, autocomplete{+}, novalidate{+}, enctype; ter : &lt;input&gt; атрибути - multiple {+}, formaction {+}, pattern {+}, title, autofocus {+}, height {+}, required {+}, checked, minlength {?}, maxlength, dirname, min {+}, formenctype {+}; ter : &lt;input&gt; атрибути - type=' file', type=' date ', type=' week', type=' datetime ', type=' email ', type=' submit', type=' hidden', type=' datetime-local '; ter : &lt;textarea&gt; атрибути - id, wrap {+}, style, name, class, autofocus {+}, minlength {+-}; ter : &lt;button&gt; атрибути - value, formtarget {+}, formnovalidate {+}, name; ter : &lt;option&gt; атрибути - value, disabled, title, class; ter : &lt;optgroup&gt; атрибути - style, title, lang; ter : &lt;select&gt; атрибути - form {+}, required {+}, disabled, autofocus {+}, style; ter : &lt;isindex&gt;{-} атрибути - title, style, action;</p>

	ter : <keygen> атрибути - id, challenge, title, lang, name; ter : <summary> атрибути - dir, lang, style; ter : <progress> атрибути - class, value, lang; ter : <details> атрибути - open, id, title;
30	ter : <form> атрибути - enctype, class, lang, target, action, title; ter : <input> атрибути - dirname, dir, title, src, autofocus {+}, pattern {+}, width {+}, formenctype {+}, class, name, height {+}, maxlength, height {+}; ter : <input> атрибути - type='radio', type='text', type='tel', type='range', type='password', type='hidden', type='datetime', type='image'; ter : <textarea> атрибути - dirname, form {+}, placeholder {+}, readonly, lang, class, disabled; ter : <button> атрибути - formenctype {+}, formaction {+}, autofocus {+}, formmethod {+}; ter : <select> атрибути - required {+}, form {+}, class, lang, style; ter : <isindex>{-} атрибути - action, dir, prompt; ter : <fieldset> атрибути - lang, style, id; ter : <option> атрибути - value, id, lang, label; ter : <output> атрибути - name, lang, id, title; ter : <progress> атрибути - style, max, value; ter : <details> атрибути - dir, title, style; ter : <datalist> атрибути - lang, id, title;
31	ter : <form> атрибути - dir, novalidate{+}, id, target, accept{-}, autocomplete{+}; ter : <input> атрибути - checked, size, required {+}, form {+}, disabled, id, height {+}, formmethod {+}, width {+}, formnovalidate {+}, alt, dir, name; ter : <input> атрибути - type='url', type='color', type='button', type='submit', type='hidden', type='file', type='password', type='tel'; ter : <textarea> атрибути - autofocus {+}, class, title, readonly, form {+}, wrap {+}, cols; ter : <button> атрибути - form {+}, autofocus {+}, formnovalidate {+}, disabled; ter : <isindex>{-} атрибути - style, action, class; ter : <option> атрибути - lang, label, style, dir; ter : <fieldset> атрибути - title, style, lang; ter : <legend> атрибути - title, id, dir; ter : <details> атрибути - open, style, lang; ter : <keygen> атрибути - challenge, disabled, name, class, style; ter : <output> атрибути - name, style, lang, form; ter : <meter> атрибути - min, value, style, max, lang;

## Додаток

### 10. Регулярні вирази (додатково)

Регулярний вираз (в програмуванні) — це рядок, що описує або збігається з множиною рядків, відповідно до набору спеціальних синтаксичних правил. Вони використовуються в багатьох текстових редакторах та допоміжних інструментах для пошуку та зміни тексту на основі заданих шаблонів. Багато мов програмування підтримують регулярні вирази для роботи з рядками. Наприклад, Perl та Tcl мають потужний механізм для роботи, вбудований безпосередньо в їх синтаксис. Завдяки набору утиліт (включаючи редактор sed та фільтр grep), що входили до складу дистрибутивів Юнікс регулярні вирази стали відомими та поширеними.

Синтаксис регулярних виразів залежить від інтерпретатора, що використовується для їхньої обробки. Однак, із незначними відхиленнями, майже всі поширені інтерпретатори регулярних виразів мають спільні правила.

#### Звичайні символи (літерали) і спеціальні символи (метасимволи)

Найпростішим регулярним виразом, з якого формуються складні, є звичайний символ. Більшість символів у регулярному виразі представляють самі себе за винятком спеціальних символів (метасимволів) [ ] \ ^ \$ . | ? \* + ( ) { }, яким може передувати символ \ (зворотна коса риска), котра робить метасимволи «екранованими», «захищеними» для представлення, відображення їх самих як символів тексту. Можна екранувати деяку послідовність символів, розмістивши її між \Q і \E.

#### Приклад Відповідність

```
a\.*      a. або a
a\\*\\b   a\\b
a\[F\]    a[F]
\Q+--*/\E +--*/
```

Аналогічно можуть бути представлені інші спеціальні символи (набір символів, що вимагають екранування, може відрізнитися залежно від конкретної реалізації). Частина символів, які в тій або іншій реалізації не вимагають екранування (наприклад, кутові дужки < >), можуть бути екрановані з міркувань зручності читання.

Залежно від інтерпретатора регулярних виразів, метасимволи «?», «+», «{», «|», «(», та «)» можуть втрачати своє спеціальне значення, замість цього слід вживати «\?», «\+», «\{», «\|», «\(», та «\)».

### Будь-який символ

Метасимвол `.` (крапка) означає один будь-який символ, але в деяких реалізаціях окрім символу нового рядка.

### Символьні класи (набори символів)

Набір символів у квадратних дужках `[ ]` іменується символьним класом і дозволяє вказати інтерпретаторові регулярних виразів, що на даному місці в рядку може стояти один із перерахованих символів. Зокрема, `[абв]` задає можливість появи в тексті одного із трьох зазначених символів, а `[1234567890]` задає відповідність одній із цифр.

Можливе зазначення діапазонів символів: наприклад, `[А-Яа-я]` відповідає всім літерам російського алфавіту, за винятком літер «Ё» і «ё».

Якщо потрібно вказати символи, які не входять у зазначений набір, то використовують символ `^` усередині квадратних дужок, наприклад, `[^0-9]` означає будь-який символ, крім цифр.

Додавання в набір спеціальних символів шляхом екранування символом «\» — найпростіший спосіб. Деякі символьні класи можна замінити спеціальними метасимволами:

Символ	Опис
<code>\d</code>	Відповідає цифрі. Еквівалентно <code>[0-9]</code>
<code>\D</code>	Відповідає нецифровому символу. Еквівалентно <code>[^0-9]</code>
<code>\s</code>	Відповідає будь-якому пробільному символу. Еквівалентно <code>[\f\n\r\t\v]</code>
<code>\S</code>	Відповідає будь-якому непробільному символу. Еквівалентно <code>[^\f\n\r\t\v]</code>
<code>\w</code>	Відповідає будь-якому літерному символу, цифровому й знаку підкреслення. Еквівалентно <code>[:word:]</code>
<code>\W</code>	Відповідає будь-якому символу, крім літерного символу, цифрового або підкреслення. Еквівалентно <code>[^:word:]</code>

### Позиція всередині рядка

Наступні символи дозволяють позиціонувати регулярний вираз щодо елементів тексту: початку й кінця рядка, меж слова.

Представлення	Позиція	Приклад	Відповідність
<code>^</code>	Початок рядка	<code>^a</code>	<code>aaa aaa</code>
<code>\$</code>	Кінець рядка	<code>a\$</code>	<code>aaa aa</code>
<code>\b</code>	Межа слова	<code>a\b</code>	<code>aaa aa</code>
		<code>\ba</code>	<code>aaa aa</code>
<code>\B</code>	Не межа слова	<code>\Ba\B</code>	<code>aaa aa</code>
<code>\G</code>	Попередній успішний пошук	<code>\Ga</code>	<code>aaa aaa</code> (пошук зупинився на 4-й позиції — там, де не знайшлося <code>a</code> )

### Квантифікація (пошук послідовностей)

*Квантифікатор* після символу, символьного класу або групи визначає, скільки разів попередній вираз може зустрічатися. Варто враховувати, що квантифікатор може стосуватися більш ніж до одного символу в регулярному виразі, тільки якщо це символьний клас або група.

Представлення	Кількість повторень	Приклад	Відповідність
{ <i>n</i> }	Рівно <i>n</i> разів	colou{3}r	colouuur
{ <i>m</i> , <i>n</i> }	Від <i>m</i> до <i>n</i> включно	colou{2,4}r	colouur, colouuur, colouuuur
{ <i>m</i> , }	Не менше <i>m</i>	colou{2,}r	colouur, colouuur, colouuuur і т.д.
{, <i>n</i> }	Не більше <i>n</i>	colou{,3}r	color, colour, colouur, colouuur

Представлення	Кількість повторень	Еквівалент	Приклад	Відповідність
*	Нуль або більше	{0, }	colou*r	color, colour, colouur і т.д.
+	Одне або більше	{1, }	colou+r	colour, colouur і т.д. (але не color)
?	Нуль або одне	{0, 1}	colou?r	color, colour

Часто використовується послідовність `.*` для позначення будь-якої кількості будь-яких символів між двома частинами регулярного виразу.

Символьні класи в поєднанні із квантифікаторами дозволяють установлювати відповідності з реальними текстами. Наприклад, колонками цифр, телефонами, поштовими адресами, елементами HTML-розмітки й ін.

Якщо символи { } не утворюють квантифікатор, їхнє спеціальне значення ігнорується.

## Жадібна й ледача квантифікація

### Приклад використання жадібних і ледачих виразів

Вираз (`<.*>`) відповідає рядку, що містить декілька тегів HTML-розмітки, цілком.

```
<p><b>Вікіпедія</b> - вільна енциклопедія, у якій <i>кожен</i> може змінити або доповнити будь-яку статтю</p>.
```

Щоб виділити окремі теги, можна застосувати ледачу версію цього виразу: (`<.*?>`) Їй відповідає не весь показаний вище рядок, а окремі теги (виділені кольором):

```
<p><b>Вікіпедія</b> - вільна енциклопедія, у якій <i>кожен</i> може змінити або доповнити будь-яку статтю</p>.
```

У деяких реалізаціях квантифікаторам у регулярних виразах відповідає максимально довгий рядок із можливих (квантифікатори є *жадібними*, англ. *greedy*). Це може стати значною проблемою. Наприклад, часто очікують, що вираз (`<.*>`) знайде в тексті [теги](#) HTML. Однак, якщо в тексті є більше одного HTML-тегу, то цьому виразу відповідає цілком рядок, що містить множину тегів.

```
<p><b>Вікіпедія</b> - вільна енциклопедія, у якій <i>кожен</i> може змінити або доповнити будь-яку статтю</p>.
```

Цю проблему можна вирішити двома способами.

1. Ураховувати символи, що *не* відповідають бажаному взірцю (`<[>]*>` для вищеописаного випадку).
2. Визначити квантифікатор як *нежадібний* (ледачий, англ. *lazy*) — більшість реалізацій дозволяють це зробити, додавши після нього знак питання. Використання ледачих квантифікаторів може викликати зворотну проблему, коли виразу відповідає занадто короткий, зокрема, порожній рядок.

### Жадібний Ледачий

```
*      *?
+      +?
{n, }  {n, }?
```

Також спільною проблемою як жадібних, так і ледачих виразів є точки повернення для перебору варіантів виразу. Точки ставляться після кожної ітерації квантифікатора. Якщо інтерпретатор не знайшов відповідності після квантифікатора, то він починає повертатися за всіма встановленими точками, перераховуючи звідти вираз по-іншому.

## Ревнива квантифікація (Найжадібніша)

При пошуку виразу `(a++)+b` у рядку `aaaaa` інтерпретатор піде приблизно таким шляхом:

1. `aaaaa`

2. aaaaa
3. aaaaa
4. aaaaa
5. aaaaa
6. aaaaa
7. aaaaa — і тільки тут, перевіривши всі точки повернення, здається.

При використанні ревнивого квантифікатора буде виконаний тільки перший крок алгоритму.

На відміну від звичайної (жадібної) квантифікації, ревнива квантифікація не тільки намагається знайти максимально довгий варіант, але ще й не дозволяє алгоритму вертатися до попередніх кроків пошуку для того, щоб знайти можливі відповідності для частини регулярного виразу, що залишилася.

Використання ревливих квантифікаторів збільшує швидкість пошуку, особливо в тих випадках, коли рядок не відповідає регулярному виразу. Крім того, ревливі квантифікатори можуть бути використані для виключення небажаних збігів.

### Жадібний Ревливий

*	*+
?	?+
+	++
{n, }	{n, }+

#### Приклад

#### Відповідність

ab(xa)\*+a abxaabxaa; але не abxaabxaa, тому що літера **a** уже зайнята

### Групування

#### Позначення групи

Круглі дужки використовуються для визначення області дії й пріоритету операцій.

Шаблон усередині групи обробляється як єдине ціле й може бути квантифікованим.

Наприклад, вираз (тр[ay] м-?)\* знайде послідовність виду трумтрам-трум-трамтрум.

#### Зворотний зв'язок

Одне із застосувань групування — повторне використання раніше знайдених груп символів (*підрядків, блоків, позначених підвиразів*). При обробці виразу підрядки, що знайдені за шаблоном усередині групи, зберігаються в окремій області пам'яті й отримують номер, починаючи з одиниці. Кожному підрядку відповідає пара дужок у регулярному виразі. Квантифікація групи не впливає на збережений результат, тобто зберігається лише перше входження. Зазвичай підтримується до 9 нумерованих підрядків із номерами від 1 до 9, але деякі інтерпретатори дозволяють працювати з більшою кількістю. Згодом у межах даного регулярного виразу можна використати позначення від \1 до \9 для перевірки на збіг із раніше знайденим підрядком.

Наприклад, регулярний вираз (та|ту)-\1 знайде рядок та-та або ту-ту, але пропустить рядок та-ту.

Також раніше знайдені підрядки можна використовувати при заміні за регулярним виразом. У такому разі в текст, що заміщає, вставляються ті ж позначення, що й у межах самого виразу.

#### Групування без зворотного зв'язку

Якщо група використовується тільки для групування і її результат надалі не буде потрібен, то можна використати групування виду (? :шаблон). Під результат такого групування не виділяється окрема область пам'яті й, відповідно, їй не призначається номер. Це позитивно впливає на швидкість виконання виразу, але знижує зручність читання.

#### Атомарне групування

Атомарне групування (виду (?>шаблон)), так само як групування без зворотного зв'язку, не створює зворотних зв'язків. На відміну від нього, таке групування забороняє вертатися назад по рядку, якщо частина шаблону вже знайдена.

Приклад	Відповідність	Створювані групи
<code>a (bc b x) cc</code>	<code>abccaxcc</code> <code>abccaxcc</code>	<code>abccaxcc</code> <code>abccaxcc</code>
<code>a (? :bc b x) cc</code>	<code>abccaxcc</code>	
<code>a (?&gt;bc b x) cc</code>	але не <code>abccaxcc</code> : варіант <code>x</code> знайдений, інші зігноровані	немає
<code>a (?&gt;x*) xa</code>	не знайдеться <code>axxxa</code> : усі <code>x</code> зайняті, і немає повернення всередину групи	

Атомарне групування виконується ще швидше, ніж групування без зворотного зв'язку, і зберігає процесорний час при виконанні решти виразу, тому що забороняє перевірку будь-яких інших варіантів усередині групи, коли один варіант уже знайдений. Це дуже корисно при оптимізації груп із множиною різних варіантів.

## Модифікатори

Модифікатори діють із моменту входження й до кінця регулярного виразу або протилежного модифікатора. Деякі інтерпретатори можуть застосувати модифікатор до всього виразу, а не з моменту його входження.

Синтаксис	Опис
<code>(?i)</code>	Включає нечутливість виразу до регістра символів (англ. <i>case insensitivity</i> )
<code>(?-i)</code>	Виключає
<code>(?s)</code>	Включає режим відповідності точки символам переносу
<code>(?-s)</code>	Виключає рядка й повернення каретки
<code>(?m)</code>	Символи <code>^</code> і <code>\$</code> викликають після й до символів нового рядка
<code>(?-m)</code>	відповідність тільки із початком і кінцем рядка
<code>(?x)</code>	Включає режим без урахування пробілів між частинами регулярного виразу й дозволяє використовувати <code>#</code> для коментарів
<code>(?-x)</code>	Виключає

Групи-модифікатори можна об'єднувати в одну групу: `(?i-sm)`. Така група включає режим `i` і виключає режим `s`, `m`. Якщо використання модифікаторів потрібне тільки в межах групи, то потрібний шаблон вказується всередині групи після модифікаторів і двокрапки. Наприклад, `(?-i)(?i:tv) set` знайде `TVset`, але не `TVSET`.

## Коментарі

Для додавання коментарів у регулярний вираз можна використовувати групи-коментарі виду `(?#коментар)`. Така група інтерпретатором цілком ігнорується й не перевіряється на входження в текст. Наприклад, вираз `A(?#тут коментар)B` відповідає рядку `AB`.

## Перерахування

Вертикальна риса розділяє допустимі варіанти. Наприклад, `gray|grey` відповідає `gray` або `grey`. Варто пам'ятати, що перебір варіантів виконується зліва праворуч, як вони вказані.

Якщо потрібно вказати перелік варіантів усередині складнішого регулярного виразу, то його потрібно взяти в групу. Наприклад, `gray|grey` або `gr(a|e)y` описують рядок `gray` або `grey`. У разі односимвольних альтернатив кращий варіант `gr[ae]y`, тому що порівняння із символьним класом виконується простіше, ніж обробка групи з перевіркою на всі її можливі модифікатори й генерацією зворотного зв'язку.

## Перегляд уперед та назад

У більшості реалізацій регулярних виразів є спосіб провадити пошук фрагмента тексту, «переглядаючи» (але не включаючи в знайдене) навколишній текст, що розташований



до або після шуканого фрагмента тексту. Наприклад, таким способом легко знайти ім'я теґу HTML, не включаючи в результат пошуку оточуючі його кутові дужки або інші знаки, але й, не упускаючи їх «з уваги» при пошуку потрібного контексту. Перегляд із запереченням використовується рідше й «стежить» за тим, щоб указані відповідності, навпаки, не зустрічалися до або після шуканого текстового фрагмента.

Представлення	Вид перегляду	Приклад	Відповідність
(?=шаблон)	Позитивний перегляд уперед	Людовик (?=XVI)	ЛюдовикXV, ЛюдовикXVI, ЛюдовикXVIII, ЛюдовикLXVII, ЛюдовикXXL
(?!шаблон)	Негативний перегляд уперед (із запереченням)	Людовик (?!XVI)	ЛюдовикXV, ЛюдовикXVI, ЛюдовикXVIII, ЛюдовикLXVII, ЛюдовикXXL
(?<=шаблон)	Позитивний перегляд назад	(?<=Сергій) Іванов	Сергій Іванов, Ігор Іванов
(?!шаблон)	Негативний перегляд назад (із запереченням)	(?!Сергій) Іванов	Сергій Іванов, Ігор Іванов

## Пошук за умовою

У багатьох реалізаціях регулярних виразів існує можливість вибирати, яким шляхом піде перевірка в тому або іншому місці регулярного виразу на підставі вже знайдених значень.

Представлення	Пояснення	Приклад	Відповідність
(? (=?якщо) то   інакше)	Якщо операція перегляду успішна, то далі виконується частина <i>то</i> , інакше виконується частина <i>інакше</i> . У виразі може використовуватися кожна із чотирьох операцій перегляду. Варто враховувати, що операція перегляду нульової ширини, тому частини <i>то</i> у разі позитивного або <i>інакше</i> у разі негативного перегляду повинні містити в собі опис шаблону з операції перегляду.	(? (?<=а) м   п)	ма <del>м</del> , па <del>п</del>
(? (п) то   інакше)	Якщо <i>n</i> -а група повернула значення, то пошук за умовою виконується за шаблоном <i>то</i> , інакше за шаблоном <i>інакше</i> .	(а) ? (? (1) м   п)	ма <del>м</del> , па <del>п</del>

