

WEB-ПРОГРАМУВАННЯ



# ХАРАКТЕРИСТИКА ЕЛЕМЕНТІВ ДЛЯ СТВОРЕННЯ ФОРМ

Форма HTML — розділ документа, що дозволяє користувачеві вводити інформацію для подальшої обробки системою.

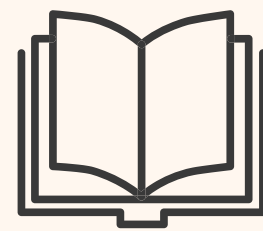
# СТВОРЕННЯ ПРОСТОЇ ФОРМИ

Тег **<form>** ... **</form>** задає початок і кінець форми

HTML-форма містить елементи форми.

Елементи форми - це різні типи елементів вводу, такі як текстові поля, прапорці, перемикачі, кнопки відправки і та інше. Окрім елементів форм там можна розташовувати будь-які тексти і майже будь-які теги HTML.

```
<form>  
.  
елементи форми  
.  
</form>
```



**HTML-форма**

# HTML елементи <form>

HTML елемент <form> може містити один або кілька таких елементів форми:



- ✓ `<input>`
- ✓ `<label>`
- ✓ `<select>`
- ✓ `<textarea>`
- ✓ `<button>`
- ✓ `<fieldset>`
- ✓ `<legend>`
- ✓ `<datalist>`
- ✓ `<output>`
- ✓ `<option>`
- ✓ `<optgroup>`

## Елемент `<input>`

Елемент `<input>` є найбільш важливим елементом форми.

Елемент `<input>` може відображатися кількома способами, в залежності від атрибута `type`.

*наприклад:*

✓  
`<input type="text">`

Визначає однорядкове поле вводу тексту

✓  
`<input type="radio">`

Визначає перемикач (для вибору одного з кількох варіантів)

✓  
`<input type="submit">`

Визначає кнопку відправки (для відправки форми)





## Введення тексту



**<input type='text'>** визначає однорядкове поле вводу для введення тексту - **text input**:

*приклад коду:*

```
<form>  
Ім'я:<br>  
<input type='text' name='firstname'><br>  
Прізвище:<br>  
<input type='text' name='lastname'>  
</form>
```

*Ось так це буде виглядати в браузері:*

Ім'я:

Прізвище:

ширина текстового поля за умовчанням містить **20** символів



## Радіо кнопка введення



**`<input type='radio'>`** визначає радіо кнопку

Перемикачі (радіо-кнопки) дозволяють користувачу обрати **ОДИН** із обмеженої кількості варіантів

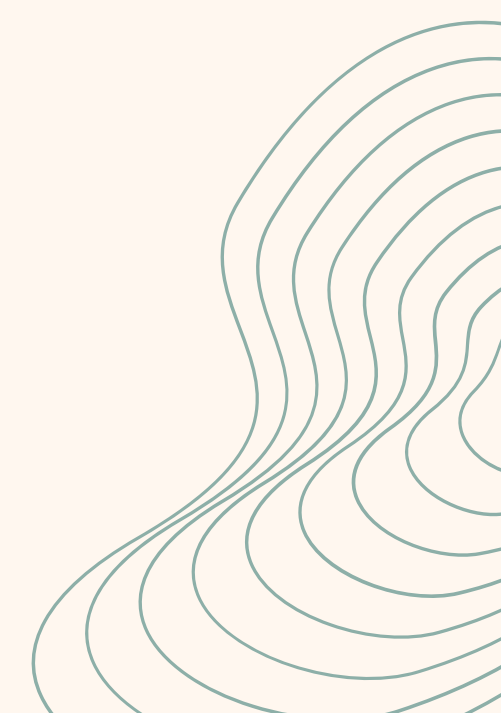
*приклад коду:*

```
<form>  
<p> Оберіть стать </p>  
  <input type='radio' name='gender'  
value='male' checked> Чоловіча<br>  
  <input type='radio' name='gender'  
value='female'> Жіноча<br>  
  <input type='radio' name='gender'  
value='other'> Інше  
</form>
```

*Ось так це буде виглядати в браузері:*

Оберіть стать

- Чоловіча
- Жіноча
- Інше





## Кнопка “Відправити”



**<input type='submit'>** визначає кнопку для відправки даних форми в обробник форм

Обробник форм зазвичай представляє собою серверну сторінку зі скриптом для обробки вхідних даних. Обробник форми вказується в атрибуті **action** форми

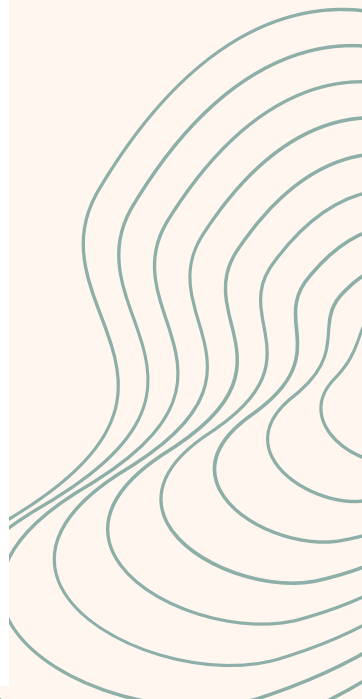
*приклад коду:*

```
<form action='/action_page.html'>  
Ім'я:<br>  
<input type='text' name='firstname'  
value='Ольга'><br>  
Прізвище:<br>  
<input type='text' name='lastname'  
value='Кобилянська'><br><br>  
<input type='submit' value='Відправити'>  
</form>
```

*Ось так це буде виглядати в браузері:*

Ім'я:

Прізвище:



# АТРИБУТ Action

Якщо атрибут **action** пропущений, дію встановлено на поточну сторінку

Атрибут **action** визначає дію, яку буде виконано при відправці форми. Зазвичай дані форми відправляються на веб-сторінку на сервері, коли користувач натискає кнопку відправки.

У наведеному на попередньому слайді прикладі дані форми відправляються на сторінку на сервері з ім'ям **"/action\_page.html"**.

Ця сторінка містить серверний скрипт, який обробляє дані форми

# АТРИБУТ Target

Атрибут **target** вказує, чи буде відправлений результат відкриватися в новій вкладці браузера, у фреймі або в поточному вікні. За замовчуванням значення **'\_self'** означає, що форма буде відправлена в поточному вікні. Щоб зробити результат форми відкритим в новій вкладці браузера, використовуйте значення **'\_blank'**

Іншими допустимими значеннями є **'\_parent'**, **'\_top'** або ім'я, яке представляє ім'я `iframe`.

приклад:

```
<form action='/action_page.html' target='_blank'>
```



# АТРИБУТ Method

Атрибут **method** вказує метод **HTTP (GET або POST)**, який буде використовуватись при відправці даних форми

## GET

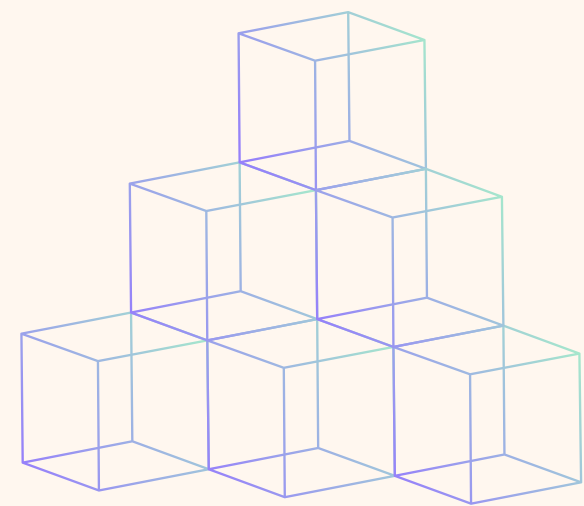
```
<form action='/action_page.html' method='get'>
```

## POST

```
<form action='/action_page.html' method='post'>
```

Головна відмінність методів POST і GET полягає у способі передачі інформації. У методі GET параметри передаються через адресний рядок, тобто по суті в HTTP заголовку запиту, в той час як у методі POST параметри передаються через тіло HTTP-запиту і ніяк не відбиваються на вигляді рядка.

---



## Коли використовувати метод GET?

Метод за замовчуванням при відправці даних форми - GET. Однак при використанні GET відправлені дані форми будуть видимі в полі адреси сторінки (адресному рядку)

### Примітка по GET:



Додає дані в формі відправки до URL-адреси в парах ім'я/значення



Довжина URL-адреси обмежена (близько 3000 символів)



Ніколи не використовуйте GET для відправки конфіденційних даних!



Корисно для подачі форм, де користувач хоче зробити закладку результату

## Коли використовувати метод POST?

Завжди використовуйте POST, якщо дані форми містять конфіденційну або особисту інформацію. Метод POST не відображає відправлені дані форми в адресному рядку сторінки.



POST не має обмежень за розмірами і може використовуватися для відправки великих обсягів даних



Відправлені з POST форми можна додати в закладки



Тег **<TEXTAREA>** використовується для того, щоб дозволити користувачеві вводити більше одного рядка інформації



Параметри тегу  
**<TEXTAREA>**



**ROWS**      **COLS**  
**WRAP**      **PHYSICAL**  
**ACCESSKEY**  
**ALIGN**      **TABINDEX**  
**DISABLED**  
**READONLY**

Параметри не є обов'язковими, але зазвичай все ж таки, слід вказати хоча б NAME, COLS і ROWS.

ROWS – Кількість рядків у полі

COLS – Кількість стовпців у полі

WRAP – Вказує чи розривати довгі рядки: OFF – не розривати; VIRTUAL – розривати на екрані, але всередині залишати їх цілими; PHYSICAL – чесно розривати

ACCESSKEY – Задає клавішу, яку потрібно натиснути спільно з Alt, щоб одразу потрапити до цього поля

ALIGN – Визначає вирівнювання елемента щодо навколишнього тексту

TABINDEX – Задає номер даного елемента у послідовності проходження елементів форми за допомогою кнопки Tab

DISABLED – Поле не дозволяє користувачу вводити інформацію, і ні на що не реагує. Використовується лише для відображення початкової (за замовчуванням) інформації

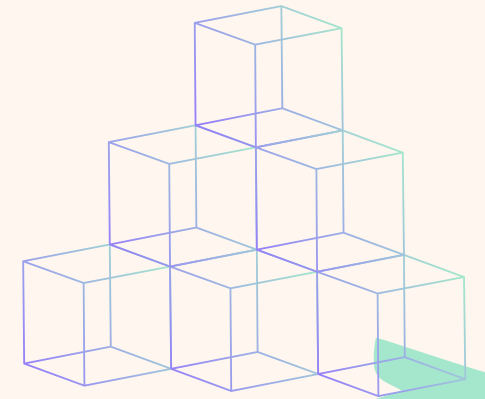
READONLY – Схоже на попереднє, але поле отримує фокус і в ньому може бути курсор. Проте його заборонено редагувати



# Приклад застосування тегу <TEXTAREA>



Запорізький національний  
університет.  
Кафедра економічної кібернетики.



Перший рядок вихідного тексту.  
Другий рядок вихідного  
тексту.

Надіслати

Сбросить

```
<TEXTAREA NAME="addr"  
ROWS=6 COLS=32>
```

Запорізький національний  
університет.  
Кафедра економічної кібернетики.

```
</TEXTAREA>
```

```
<FORM
```

```
action="http://somesite.com/prog/text-  
read" method="post">
```

```
<P> <TEXTAREA name="thetext" rows="10"  
cols="40">
```

Перший рядок вихідного тексту.  
Другий рядок вихідного  
тексту.

```
</TEXTAREA>
```

```
<BR><INPUT type="submit" value="Надіслати">
```

```
<BR><INPUT type="reset">
```

```
</P> </FORM>
```

HTML код

результат

# Приклад застосування тегу <SELECT>

Елемент <select> визначає drop-down список (випадаючий список), список з можливістю вибрати більш ніж один варіант.

## Параметри тегу <SELECT>

### SIZE

Кількість елементів у списку видимих на екрані. Якщо цей параметр більше 1, то маємо справу зі списком, інакше – зі спадаючим меню.

### MULTIPLE

Якщо зазначено, користувач може вибрати кілька елементів списку одночасно.

### SINGLE

Якщо вказано, користувач не може вибрати кілька елементів списку одночасно. Якщо не вказано ні SINGLE, ні MULTIPLE, мається на увазі SINGLE

# Приклад застосування тегу <SELECT>

Елемент <option> визначає опцію, яка може бути обрана. За замовчуванням обраний перший елемент у випадяючому списку. Щоб визначити попередньо обраний параметр, додайте атрибут **selected** до option

```
<option  
value="fiat"  
selected>Fiat  
</option>
```

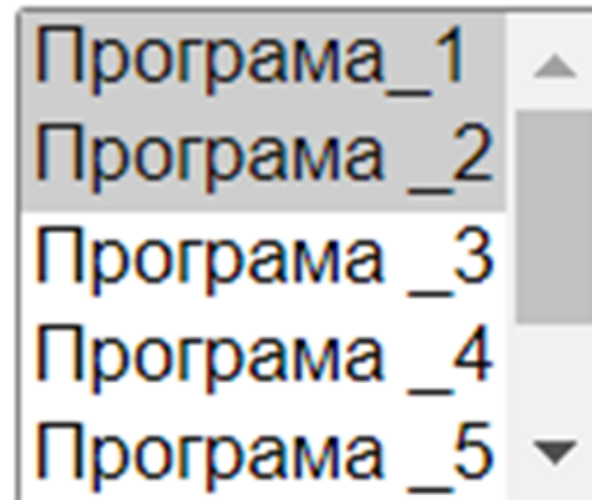
Параметр **VALUE** визначає значення для поля «за замовчуванням».

```
<select name="cars" size="4" multiple>  
<option value="volvo">Volvo</option>  
<option value="saab">Saab</option>  
<option value="fiat">Fiat</option>  
<option value="audi">Audi</option>  
</select>
```

Приклад використання атрибуту **size** (визначає кількість видимих елементів у списку) та атрибуту **multiple**, який дозволяє користувачу обрати більше одного значення зі списку

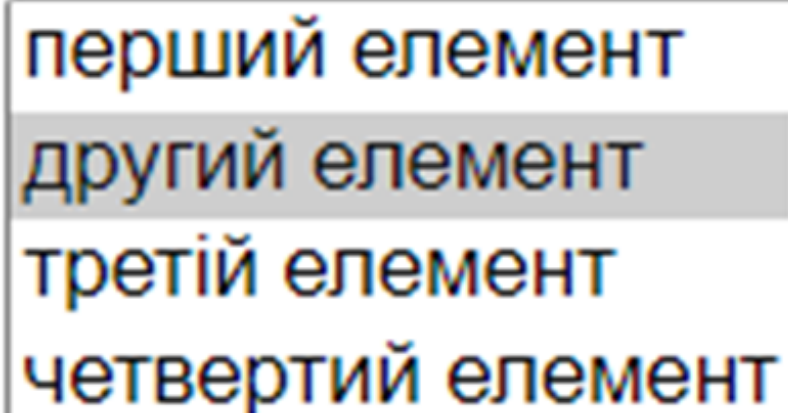
# Приклади різних елементів <SELECT>

```
<SELECT multiple size="5"  
name="component-select">  
  <OPTION selected  
value="Component_1_a">Програма_1  
</OPTION>  
  <OPTION selected  
value="Component_1_b">Програма_2  
</OPTION>  
  <OPTION>Програма_3</OPTION>  
  <OPTION>Програма_4</OPTION>  
  <OPTION>Програма_5</OPTION>  
  <OPTION>Програма_6</OPTION>  
  <OPTION>Програма_7</OPTION>  
</SELECT>
```



Програма\_1  
Програма\_2  
Програма\_3  
Програма\_4  
Програма\_5

```
<SELECT MULTIPLE NAME=group  
SIZE=4>  
  <OPTION> перший елемент  
</OPTION>  
  <OPTION SELECTED> другий  
елемент </OPTION>  
  <OPTION> третій елемент  
</OPTION>  
  <OPTION> четвертий елемент  
</OPTION>  
  <OPTION> п'ятий елемент  
</OPTION>  
</SELECT>
```

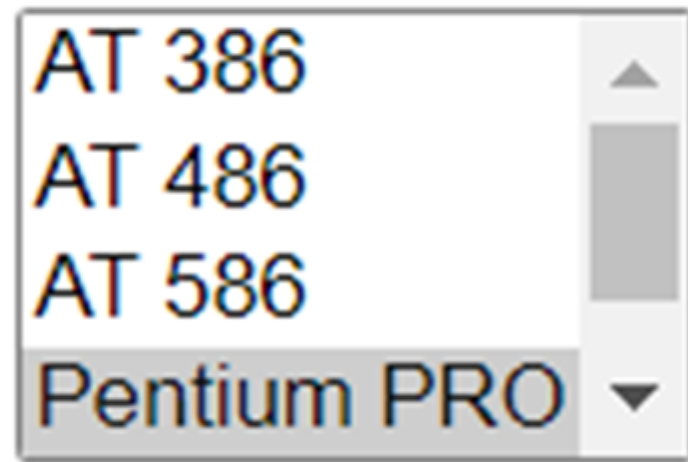


перший елемент  
другий елемент  
третій елемент  
четвертий елемент



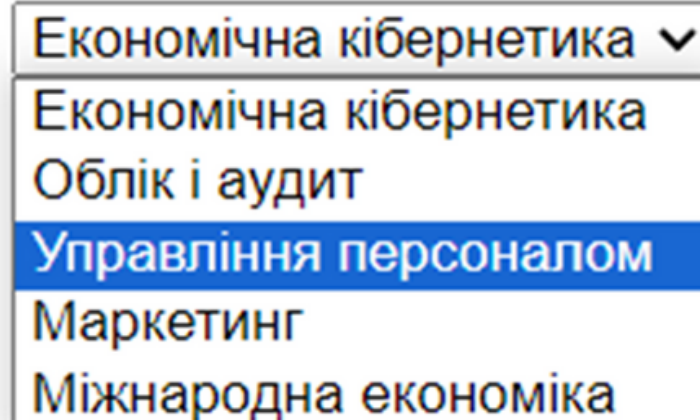
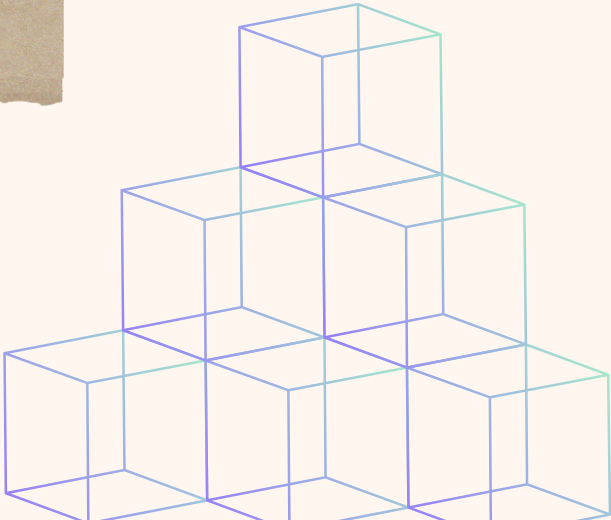
# Приклади різних елементів <SELECT>

```
SELECT NAME=group SIZE=4>  
<OPTION> AT 386 </OPTION>  
<OPTION> AT 486 </OPTION>  
<OPTION> AT 586 </OPTION>  
<OPTION SELECTED>  
Pentium PRO </OPTION>  
<OPTION> Pentium II  
</OPTION>  
</SELECT>
```



```
<SELECT NAME=group>  
<OPTION SELECTED> Економічна  
кібернетика</OPTION>  
<OPTION>Облік і аудит </OPTION>  
<OPTION> Управління персоналом  
</OPTION>  
<OPTION> Маркетинг </OPTION>  
<OPTION> Міжнародна економіка  
</OPTION>  
</SELECT>
```

Економічна кібернетика ▾



```
Економічна кібернетика ▾  
Економічна кібернетика  
Облік і аудит  
Управління персоналом  
Маркетинг  
Міжнародна економіка
```





**Елемент `<button>`  
визначає клікабельну  
кнопку**

```
<button type="button" onclick="alert ('Hello  
World!')"> Тицни мене!  
</button>
```



**HTML5 додав наступні елементи форми: `<datalist>` та `<output>`**

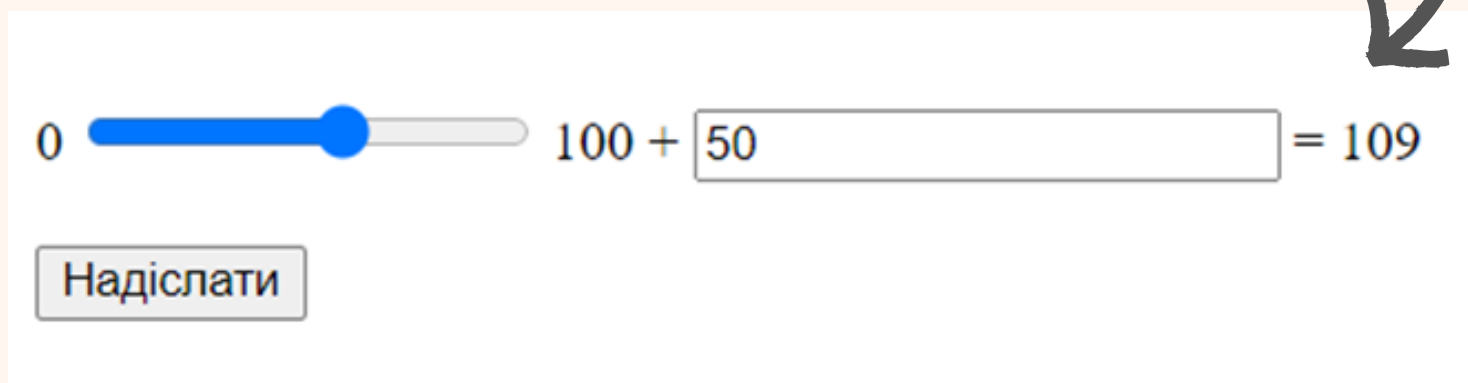
**Елемент `<datalist>` визначає список попередньо визначених параметрів для елемента `<input>`. Користувачі побачать випадаючий список попередньо визначених параметрів при введенні даних.**

Атрибут **list** елемента `<input>` повинен посилатися на атрибут **id** елемента `<datalist>`

```
<form action="/action_page.html">  
<input list="browsers">  
<datalist id="browsers">  
  <option value="Internet Explorer">  
  <option value="Firefox">  
  <option value="Chrome">  
  <option value="Opera">  
  <option value="Safari">  
</datalist>  
</form>
```

Елемент `<output>` представляє результат обрахування (наприклад, виконаний скриптом).

Виконується розрахунок і виводиться результат в елементі `<output>`



0  100 +  = 109

Надіслати

```
<form action="/action_page.html"
oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  0
  <input type="range" id="a" name="a" value="50">
  100 +
  <input type="number" id="b" name="b" value="50">
  =
  <output name="x" for="a b"></output>
  <br><br>
  <input type="submit" value="Надіслати">
</form>
```



# Приклади елементів форми



Ім'я:

Прізвище:

email:

- Вища освіта
- Середня освіта

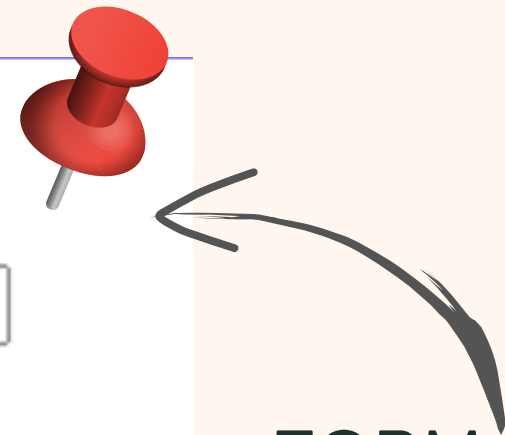
Надіслати

Очистити

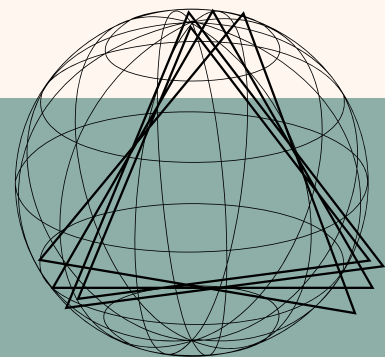
Оберіть файл:

Виберіть файл Файл не выбран

Надіслати



```
<FORM action="http://somesite.com/prog/adduser" method="post">  
<P> Ім'я: <INPUT type="text" name="firstname"><BR><br>  
Прізвище: <INPUT type="text" name="lastname"><BR><br>  
email: <INPUT type="text" name="email"><BR>  
  <INPUT type="radio" name="sex" value="Male"> Вища освіта<BR>  
  <INPUT type="radio" name="sex" value="Female"> Середня освіта<BR>  
<br>  
  <INPUT type="submit" value="Надіслати">  
<br><br><INPUT type="reset" value="Очистити">  
<BR> Оберіть файл:<BR> <INPUT TYPE="FILE" >  
<P> <INPUT TYPE="SUBMIT" VALUE="Надіслати">  
</FORM>
```



# Елементи форми



Тег	Опис
<code>&lt;form&gt;</code>	Визначає HTML форму для користувальницького вводу
<code>&lt;input&gt;</code>	Визначає елемент управління вводом
<code>&lt;textarea&gt;</code>	Визначає багаторядковий елемент управління вводом (текстова область)
<code>&lt;label&gt;</code>	Визначає мітку для елемента <code>&lt;input&gt;</code>
<code>&lt;fieldset&gt;</code>	Групує пов'язані елементи в формі
<code>&lt;legend&gt;</code>	Визначає заголовок для елемента <code>&lt;fieldset&gt;</code>
<code>&lt;select&gt;</code>	Визначає випадаючий (розкриваючий) список

# Елементи форми



Тег	Опис
<code>&lt;optgroup&gt;</code>	Визначає групу пов'язаних параметрів в випадяючому списку
<code>&lt;option&gt;</code>	Визначає параметр в випадяючому списку
<code>&lt;button&gt;</code>	Визначає клікабельну кнопку
<code>&lt;datalist&gt;</code>	Визначає список попередньо визначених параметрів для елементів управління вводом
<code>&lt;output&gt;</code>	Визначає результат розрахунку

# Список всіх атрибутів <form>:



<b>accept-charset</b>	Вказує набір символів, що використовується у відправленій формі (за замовчуванням: набір символів сторінки).
<b>action</b>	Вказує адресу ( <i>url</i> ), куди відправити форму (за замовчуванням: сторінка відправки).
<b>autocomplete</b>	Вказує, чи повинен браузер автоматично заповнювати форму (за замовчуванням: включено).
<b>enctype</b>	Визначає кодування відправлених даних (за замовчуванням: <i>url-encoded</i> ).
<b>method</b>	Вказує метод HTTP, який використовується під час відправки форми (за замовчуванням: <i>GET</i> ).
<b>name</b>	Вказує назву, яка використовується для ідентифікації форми (для використання DOM: <i>document.forms.name</i> ).
<b>novalidate</b>	Вказує, що браузер не повинен перевіряти форму.
<b>target</b>	Вказує ціль адреси в атрибуті <i>action</i> (за замовчуванням: <i>_self</i> ).