



Работа с формами в HTML

Работа с формами

Формы в html представляют один из способов для ввода и отправки данных. Все поля формы помещаются между тегами `<form>` и `</form>`.

```
<form method="post"  
action="http://localhost:8080/login.php">  
  <input name="login" />  
  <input type="submit" value="Войти" />  
</form>
```



The image shows a rendered HTML form. It consists of a single-line text input field with a light gray border and a rounded rectangular button to its right. The button has a light gray background and the text "Войти" (Login) in a dark gray font.

Для настройки форм у элемента *form* определены следующие атрибуты:

- *method*: устанавливает метод отправки данных на сервер. Допустимы два значения: *post* и *get*.
Значение *post* позволяет передать данные на веб-сервер через специальные заголовки. А значение *get* позволяет передать данные через строку запроса.
- *action*: устанавливает адрес, на который передаются данные формы
- *enctype*: устанавливает тип передаваемых данных. Он свою очередь может принимать следующие значения:
 - *application/x-www-form-urlencoded*: кодировка отправляемых данных по умолчанию
 - *multipart/form-data*: эта кодировка применяется при отправке файлов
 - *text/plain*: эта кодировка применяется при отправке простой текстовой информации

АВТОДОПОЛНЕНИЕ

Часто веб-браузеры запоминают вводимые данные, и при вводе браузеры могут выдавать список подсказок из ранее введенных слов

```
<form method="post" autocomplete="on"  
action="http://localhost:8080/login.php">  
  <input name="login" />  
  <input name="password" />  
  <input type="submit" value="Войти" />  
</form>
```



A screenshot of a web form with two input fields and a submit button. The first input field, labeled 'login', is active and has a dropdown menu showing a list of suggestions, with 'login' selected. The second input field is empty. The submit button is labeled 'Войти'.

Элементы формы

Наиболее распространенным элементом ввода является элемент *input*. Однако реальное действие этого элемента зависит от того, какое значение установлено у его атрибута *type*.

- text: обычное текстовое поле
- password: тоже текстовое поле, только вместо вводимых символов отображаются звездочки, поэтому в основном используется для ввода пароля
- radio: радиокнопка или переключатель. Из группы радиокнопок можно выбрать только одну
- checkbox: элемент флажок, который может находиться в отмеченном или неотмеченном состоянии
- hidden: скрытое поле
- submit: кнопка отправки формы
- color: поле для ввода цвета
- date: поле для ввода даты
- datetime: поле для ввода даты и времени с учетом часового пояса
- datetime-local: поле для ввода даты и времени без учета часового пояса
- email: поле для ввода адреса электронной почты
- month: поле для ввода года и месяца
- number: поле для ввода чисел
- range: ползунок для выбора числа из некоторого диапазона
- tel: поле для ввода телефона
- time: поле для ввода времени
- week: поле для ввода года и недели
- url: поле для ввода адреса url
- file: поле для выбора отправляемого файла
- image: создает кнопку в виде картинки

Кроме элемента *input* в различных модификациях есть еще небольшой набор элементов, которые также можно использовать на форме:

- `button`: создает кнопку
- `select`: выпадающий список
- `label`: создает метку, которая отображается рядом с полем ввода
- `textarea`: многострочное текстовое поле

Атрибут `name` и `value`

У всех элементов ввода можно установить атрибуты `name` и `value`. Эти атрибуты имеют важное значение. По атрибуту `name` мы можем идентифицировать поле ввода, а атрибут `value` позволяет установить значение поля ввода.

```
<form method="get"
action="index.html">
  <input type="text" name="login"
value="Tom" />
  <input type="password"
name="password" />
```

A screenshot of a web form. It consists of two text input fields and a button. The first input field contains the text "Tom". The second input field is empty. The button is labeled "Войти" (Login) and is positioned to the right of the second input field. The form is enclosed in a light gray border.

```
</form>
```


Кнопки

Кнопки представлены элементом *button*. Они обладают широкими возможностями по конфигурации. Так, в зависимости от значения атрибута *type* мы можем создать различные типы кнопок:

- *submit*: кнопка, используемая для отправки формы
- *reset*: кнопка сброса значений формы
- *button*: кнопка без какого-либо специального назначения

Если кнопка используется для отправки формы, то есть у нее установлен атрибут *type="submit"*, то мы можем задать у нее ряд дополнительных атрибутов:

- *form*: определяет форму, за которой закреплена кнопка отправки
- *formaction*: устанавливает адрес, на который отправляется форма. Если у элемента *form* задан атрибут *action*, то он переопределяется
- *formenctype*: устанавливает формат отправки данных. Если у элемента *form* установлен атрибут *enctype*, то он переопределяется
- *formmethod*: устанавливает метод отправки формы (*post* или *get*). Если у элемента *form* установлен атрибут *method*, то он переопределяется

```
<form>
  <p><input type="text" name="login" /></p>
  <p><input type="password" name="password" /></p>
  <p>
    <button type="submit" formmethod="get"
formaction="index.html">Отправить</button>
    <button type="reset">Отмена</button>
  </p>
</form>
```

The image shows a rendered version of the HTML form. It consists of two empty text input fields stacked vertically. Below the second field are two buttons: 'Отправить' (Submit) and 'Отмена' (Cancel). The buttons are light gray with rounded corners and black text.

Кроме элемента *button* для создания кнопок можно использовать элемент *input*, у которого атрибут равен submit или reset.

```
<form>
  <p><input type="text"
name="login" /></p>
  <p><input type="password"
name="password" /></p>
  <p>
    <button
type="submit">Отправить</button>
    <button
type="reset">Отмена</button>
  </p>
  <p><input type="text"
name="login" /></p>
  <p><input type="password"
name="password" /></p>
  <p>
    <input type="submit" value="
Отправить" />
    <input type="reset" value="
Отмена" />
  </p>
</form>
```

Текстовые поля

Однострочное текстовое поле создается с помощью элемента `input`, когда его атрибут `type` имеет значение `text`

```
<input type="text"  
name="login" />
```

С помощью ряда дополнительных атрибутов можно настроить текстовое поле:

- `dir`: устанавливает направление текста
- `list`: устанавливает список подсказок для ввода в поле
- `maxlength`: максимально допустимое количество символов в текстовом поле
- `pattern`: определяет шаблон, которому должен соответствовать вводимый текст
- `placeholder`: устанавливает текст, который по умолчанию отображается в текстовом поле
- `readonly`: делает текстовое поле доступным только для чтения
- `required`: указывает, что текстовое поле обязательно должно иметь значение
- `size`: устанавливает ширину текстового поля в видимых символах
- `value`: устанавливает значение по умолчанию в текстовом поле

Доступность элементов

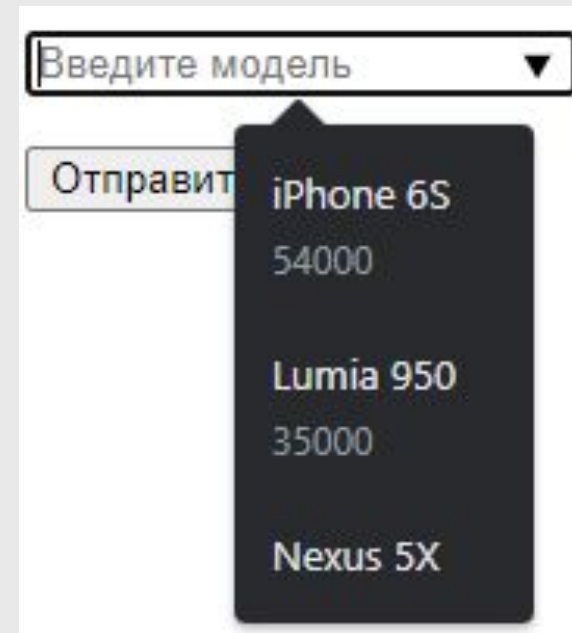
Атрибуты `readonly` и `disabled` делают текстовое поле недоступным, однако сопровождаются разным визуальным эффектом. В случае с `disabled` текстовое поле затемняется.

```
<input type="text" name="userName" value="Том"  
readonly />  
<input type="text" name="userPhone"  
value="+12345678901" disabled />
```

«Аккардеон»

Среди атрибутов текстового поля также следует отметить такой атрибут как list. Он содержит ссылку на элемент datalist, который определяет набор значений, появляющихся в виде подсказки при вводе в текстовое поле.

```
<form>
  <p><input list="phonesList" type="text" name="model" placeholder="
Введите модель" /></p>
  <p>
    <button type="submit">Отправить</button>
  </p>
</form>
<datalist id="phonesList">
  <option value="iPhone 6S" label="54000" />
  <option value="Lumia 950">35000</option>
  <option value="Nexus 5X" />
</datalist>
```



Поле поиска

Для создания полей поиска предназначен элемент `input` с атрибутом `type="search"`. Формально он представляет собой простое текстовое поле

```
<form>  
  <input type="search"  
name="term" />  
  <input type="submit"  
value="Поиск" />
```

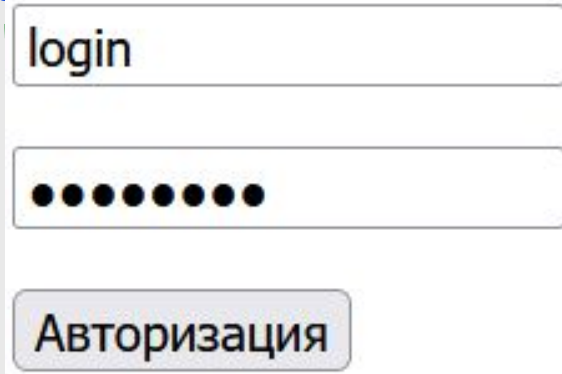


A screenshot of a search form. It consists of a white rectangular input field with a thin grey border containing the text "google". To the right of the input field is a grey button with rounded corners and the text "Поиск" in black.

Поле ввода пароля

Для ввода пароля используется элемент `input` с атрибутом `type="password"`. Его отличительной чертой является то, что вводимые символы маскируются точками

```
<form>  
  <p><input type="text" name="login"  
></p>  
  <p><input type="password"  
name="password" /></p>  
  <input type="submit" value="Ав  
Авторизация"  
</form>
```

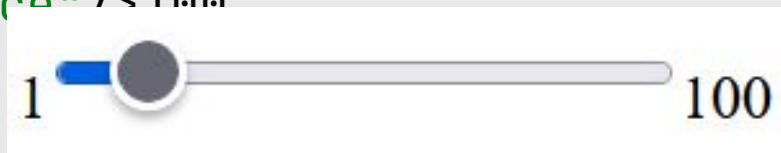


The image shows a rendered HTML form. It consists of three elements stacked vertically: a text input field containing the text "login", a password input field containing ten black dots, and a submit button with the text "Авторизация".

Ползунок

Ползунок представляет шкалу, на которой мы можем выбрать одно из значений. Для создания ползунка применяется элемент `input` с атрибутом `type="range"`. Во многом ползунок похож на простое поле для ввода чисел. Он также имеет атрибуты `min`, `max`, `step` и `value`

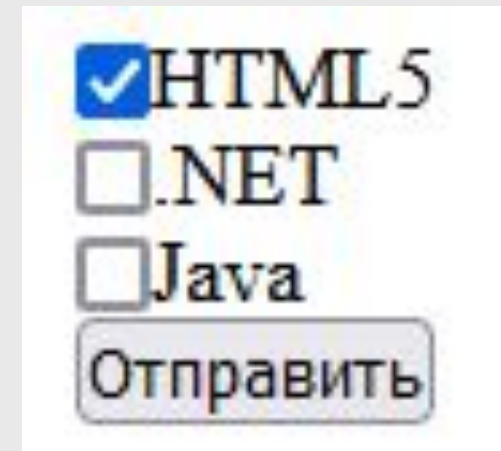
```
<input type="range" step="1" min="0" max="100" value="10"  
id="price" name="price" />
```



Чекбокс (Флажки)

Флажок представляет элемент, который может находиться в двух состояниях: отмеченном и неотмеченном. Флажок создается с помощью элемента `input` с атрибутом `type="checkbox"`

```
<form>
  <p>
    <input type="checkbox" checked
name="html5" />HTML5
  </p>
  <p>
    <input type="checkbox"
name="dotnet" />.NET
  </p>
  <p>
    <input type="checkbox"
name="java" />Java
  </p>
  <p>
    <button
type="submit">Отправить </button>
```



The image shows a rendered HTML form. It contains three checkboxes stacked vertically. The first checkbox is checked and is followed by the text 'HTML5'. The second checkbox is unchecked and is followed by the text '.NET'. The third checkbox is unchecked and is followed by the text 'Java'. Below the checkboxes is a button with the text 'Отправить' (Submit).

Радиокнопки (Переключатели)

Переключатели или радиокнопки похожи на флажки, они также могут находиться в отмеченном или неотмеченном состоянии. Только для переключателей можно создать одну группу, в которой одновременно можно выбрать только один переключатель.

```
<form>
  <h2>Укажите пол</h2>
  <p>
    <input type="radio" value="man" checked
name="gender" /> мужской
  </p>
  <p>
    <input type="radio" value="woman"
name="gender" /> женский
  </p>
</form>
```

Укажите пол

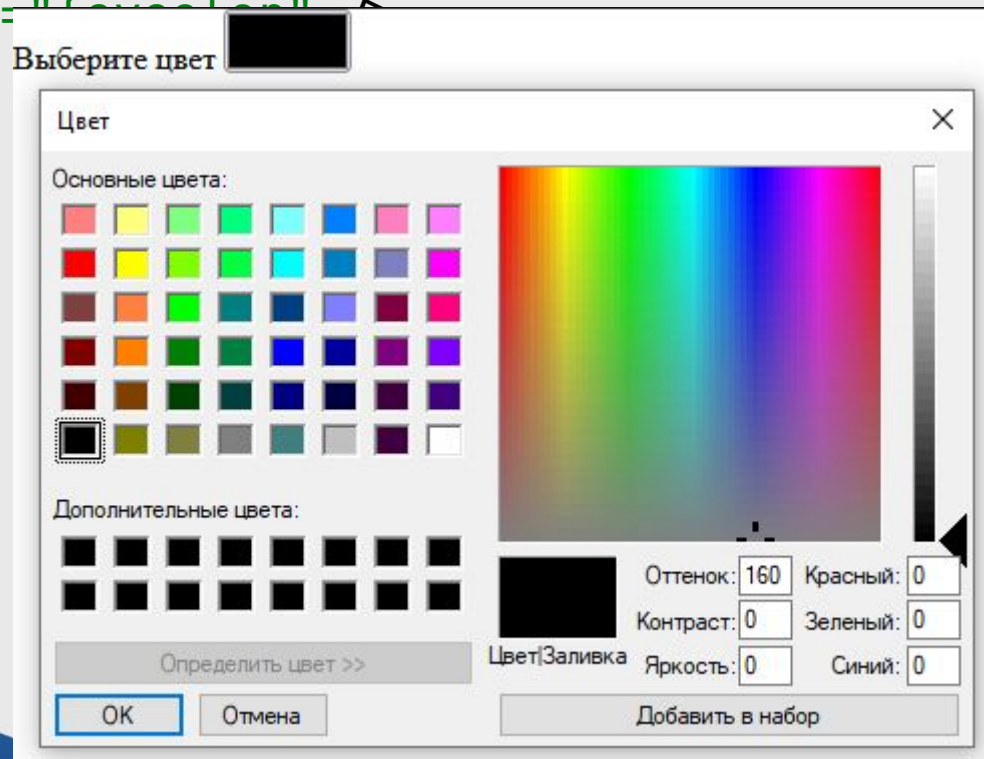
мужской

женский

Установка цвета

За установку цвета в HTML5 отвечает специальный элемент input с типом color

```
<label for="favcolor">Выберите цвет</label>  
<input type="color" id="favcolor"  
name="favcolor" />
```



Поля для ввода url, email, телефона

Ряд полей input предназначены для ввода таких данных, как url, адреса электронной почты и телефонного номера. Они однотипны и во многом отличаются только тем, что для атрибута type принимают соответственно значения email, tel и url.

Для их настройки мы можем использовать те же атрибуты, что и для обычного текстового поля:

- `maxlength`: максимально допустимое количество символов в поле
- `pattern`: определяет шаблон, которому должен соответствовать вводимый текст
- `placeholder`: устанавливает текст, который по умолчанию отображается в поле
- `readonly`: делает текстовое поле доступным только для чтения
- `required`: указывает, что текстовое поле обязательно должно иметь значение
- `size`: устанавливает ширину поля в видимых символах
- `value`: устанавливает значение по умолчанию для поля
- `list`: устанавливает привязку к элементу `datalist` со списком возможных значений

```
<form>
  <label for="email">Email: </label>
  <input id="email" name="email"
placeholder="user@gmail.com" type="email" />
  <label for="url">URL: </label>
  <input id="url" name="url" type="url" />
  <label for="phone">Телефон: </label>
  <input id="phone" name="phone"
placeholder="(XXX)-XXX-XXXX" type="tel" />
  <button type="submit">Отправить</button>
```

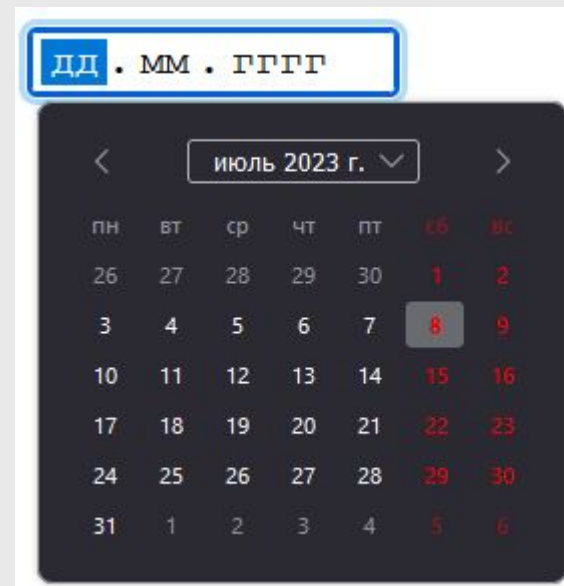
Email: URL: Телефон:

Элементы для ввода даты и времени

Для работы с датами и временем в HTML5 предназначено несколько типов элементов input:

- datetime-local: устанавливает дату и время
- date: устанавливает дату
- month: устанавливает текущий месяц и год
- time: устанавливает время
- week: устанавливает текущую неделю

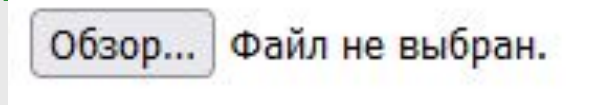
```
<input type="date" id="date"  
name="date" />
```



Отправка файлов

За выбор файлов на форме отвечает элемент `input` с атрибутом `type="file"`

```
<input type="file"  
name="file" />
```



Важно отметить, что для отправки файла на сервер форма должна иметь атрибут `enctype="multipart/form-data"`.

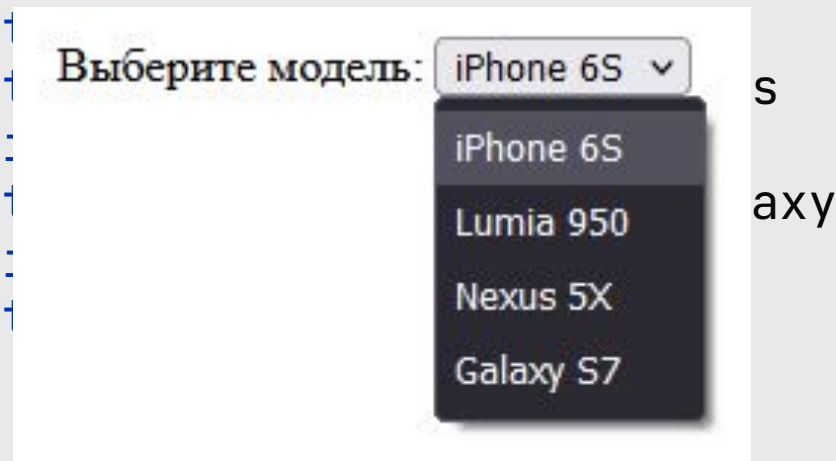
С помощью ряда атрибутов мы можем дополнительно настроить элементы выбора файла:

- `accept`: устанавливает тип файл, которые допустимы для выбора
- `multiple`: позволяет выбирать множество файлов
- `required`: требует обязательной установки файла

Список select

Элемент `select` создает список. В зависимости от настроек это может быть выпадающий список для выбора одного элемента, либо раскрытый список, в котором можно выбрать сразу несколько элементов.

```
<label for="phone">Выберите  
модель:</label>  
<select id="phone" name="phone">  
  <option value="iphone 6s">iPhone  
6S</option>  
  <option value="lumia 950">Lumia  
950</option>  
  <option value="nexus 5x">Nexus  
5X</option>  
  <option value="galaxy s7">Galaxy  
S7</option>  
</select>
```



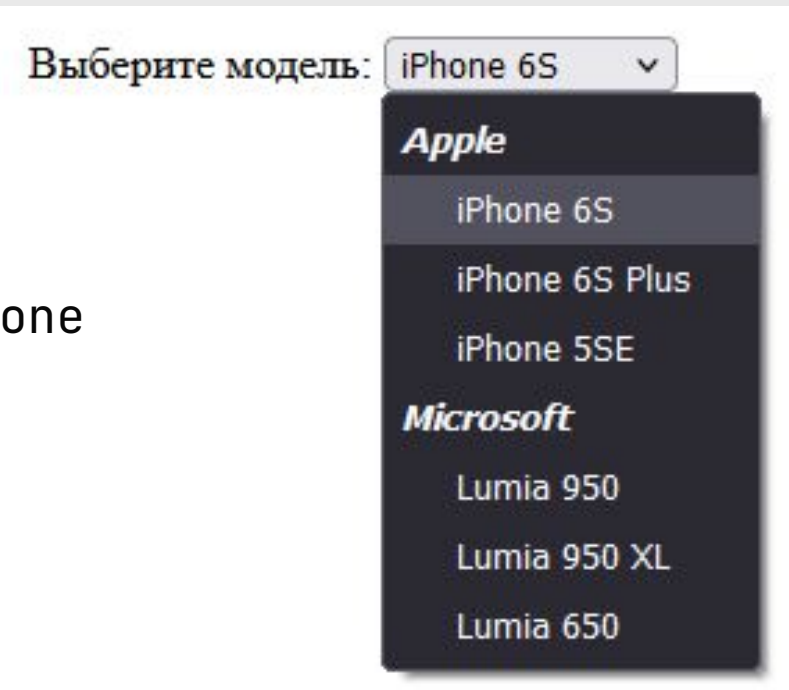
С помощью другого атрибута disabled можно запретить выбор определенного элемента. Как правило, элементы с этим атрибутом служат для создания заголовков



Select также позволяет группировать элементы с помощью тега <optgroup>

```
<label for="phone">Выберите модель:</label>
```

```
<select id="phone" name="phone">  
  <optgroup label="Apple">  
    <option value="iphone 6s">iPhone  
6S</option>  
    <option value="iphone 6s plus">iPhone  
6S Plus</option>  
    <option value="iphone 5se">iPhone  
5SE</option>  
  </optgroup>  
  <optgroup label="Microsoft">  
    <option value="lumia 950">Lumia  
950</option>  
    <option value="lumia 950 xl">Lumia 950  
XL</option>  
    <option value="lumia 650">Lumia  
650</option>
```



Textarea

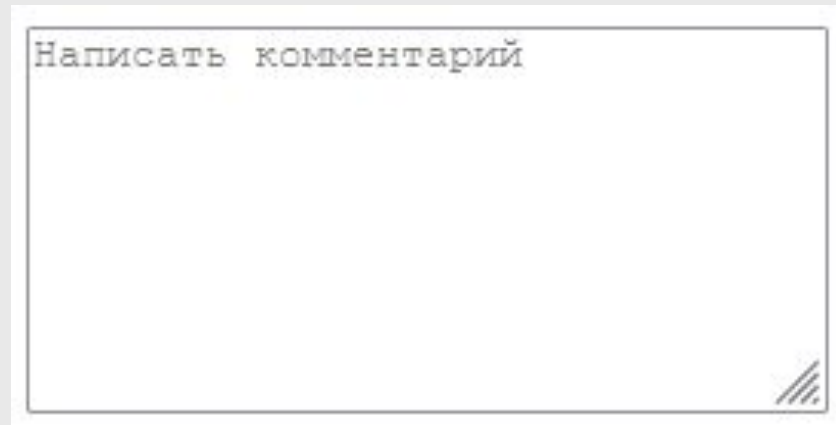
Элемент `<input type="text" />` позволяет создавать простое однострочное текстовое поле. Однако возможностей этого элемента по вводу текста бывает недостаточно, и в этой ситуации мы можем использовать многострочное текстовое поле, представленное элементом `textarea`

```
<textarea name="comment" id="comment" placeholder="Не более 200 символов"
maxlength="200"></textarea>
```

A screenshot of a text area input field. The field is rectangular with a thin border. Inside, the text "Не более 200 символов" is displayed in a light gray font, serving as a placeholder. In the bottom right corner of the field, there is a small icon consisting of three diagonal lines.

С помощью дополнительных атрибутов cols и rows можно задать соответственно количество столбцов и строк

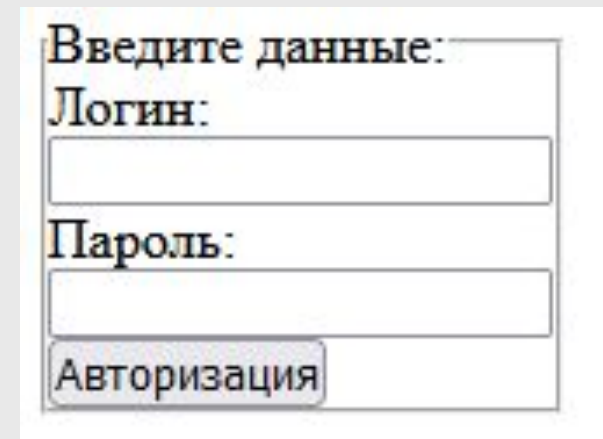
```
<textarea name="comment" placeholder="Написать комментарий" cols="30"  
rows="7"></textarea>
```

A screenshot of a web browser showing a text area. The text area has a light gray border and contains the placeholder text "Написать комментарий" in a light gray font. In the bottom right corner of the text area, there is a small icon consisting of three diagonal lines.

Элементы `fieldset` и `legend`

Для группировки элементов формы нередко применяется элемент `fieldset`. Он создает границу вокруг вложенных элементов, как бы создавая из них группу. Вместе с ним используется элемент `legend`, который устанавливает заголовок для группы элементов

```
<form>
  <fieldset>
    <legend>Введите данные:</legend>
    <label for="login">Логин:</label><br>
    <input type="text" name="login" id="login"
  /><br>
    <label for="password">Пароль:</label><br>
    <input type="password" name="password"
  id="password" /><br>
    <input type="submit" value="Авторизация">
  </fieldset>
</form>
```



Домашняя работа

- Повторить всё что было в презентации