



[Главная](#) » [HTML и HTML5](#) » [1.11. Семантические элементы HTML5](#) » [1.11.10 Формы](#)

## 1.11.10 Формы

Опубликовано: 18 января 2022

Обновлено: 5 июня 2022

0



Форма — это компонент веб-страницы с элементами управления, такими как текстовые поля, кнопки, флажки, диапазон или поле выбора цвета. Пользователь может взаимодействовать с такой формой, предоставляя данные, которые затем могут быть отправлены на сервер для дальнейшей обработки (например, возвращая результаты поиска или вычислений).

Написание формы состоит из нескольких шагов, которые могут выполняться в любом порядке: написание пользовательского интерфейса, реализация обработки на стороне сервера и настройка пользовательского интерфейса для связи с сервером.

Отправка формы серверу осуществляется чаще всего в виде запросов HTTP GET или POST. Для точного указания способа используется атрибут `method` в элементе `<form>`; способ кодирования данных формы задается с помощью атрибута `enctype`; url-адрес обработчика отправленных данных указывается в атрибуте `action`.

### Объявление скрыто

Мы используем ваши ответы, чтобы подбирать для вас подходящую рекламу



---

Сценарии на стороне клиента требуются не всегда, так как новые атрибуты форм позволяют решать некоторые задачи без участия JavaScript, например:

- Проверка пользовательского ввода перед отправкой формы с помощью атрибута `required`.
- Включение автоматического заполнения полей формы в большинстве браузеров с помощью атрибута `autocomplete`.
- Предоставление пользователю подходящего режима ввода на мобильных устройствах с помощью атрибута `inputmode`.

## HTML-элементы формы

### СОДЕРЖАНИЕ:

1. Элемент `<form>`
2. Элемент `<label>`
3. Элемент `<input>`
4. Элемент `<button>`
5. Элемент `<select>`
6. Элемент `<datalist>`
7. Элемент `<optgroup>`
8. Элемент `<option>`
9. Элемент `<textarea>`
10. Элемент `<output>`
11. Элемент `<progress>`
12. Элемент `<meter>`
13. Элемент `<fieldset>`
14. Элемент `<legend>`
15. Атрибут `autocomplete`

### 1. Элемент `<form>`



**Категории контента:** потоковое содержимое, видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается потоковое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 1. АТРИБУТЫ ЭЛЕМЕНТА <FORM>

Атрибут	Описание, принимаемое значение
accept-charset	<p>Определяет кодировку символов, которая должна использоваться для отправки формы. Если указано, значение должно соответствовать кодировке UTF-8 без учета регистра ASCII.</p> <p><b>Синтаксис:</b> accept-charset="UTF-8"</p>
action	<p>Указывает url-адрес обработчика формы. Когда форма отправляется, данные в форме преобразуются в структуру в соответствии с указанным типом кодировки enctype , а затем отправляются в место, указанное action , с использованием метода отправки данных method .</p> <p><b>Синтаксис:</b> action="subscribe.php"</p>
autocomplete	<p>Указывает, может ли значение элементов &lt;input&gt; автоматически заполняться браузером.</p> <p><b>Разрешенные значения:</b></p> <p>off — пользователь должен явно вводить значение в это поле для каждого использования.</p> <p>on — браузер может автоматически дополнять значение на основе значений, которые пользователь ввел во время предыдущих использований.</p> <p><b>Синтаксис:</b> autocomplete="off"</p>
enctype	<p>Указывает MIME-тип данных формы для отправки на сервер только в случае method="post" . Это значение может быть переопределено атрибутом formenctype в элементах &lt;button&gt; , &lt;input type="image"&gt; или &lt;input type="submit"&gt; .</p> <p><b>Разрешенные значения:</b></p> <p>application/x-www-form-urlencoded — значение по умолчанию. Данные формы кодируются как пары имя-значение, аналогично строке запроса URI. Это формат по умолчанию для POST. Тело HTTP-сообщения, отправляемого на сервер, представляет собой одну строку запроса — пары имя-значение разделены &amp; , а имена отделены от значений знаком =</p> <p>multipart/form-data — данные формы не кодируются. Это значение необходимо использовать для формы, содержащей элементы, управляющие загрузкой файлов. При отправке данных формы браузер разделяет каждый файл или вложение с помощью «границы из нескольких частей», которая является уникальным идентификатором, определяющим начало и конец каждой части. Границы части обозначаются линиями, начинающимися с тире. Это позволяет отправлять несколько частей в одном запросе и идентифицировать каждую из них со своими собственными метаданными, такими как MIME-тип, имя файла и т.д.</p> <p>text/plain — символы не кодируются, а пробелы заменяются на символ + . Полезен только для отладки.</p>

	<b>Синтаксис:</b> <code>enctype="application/x-www-form-urlencoded"</code>
method	<p>Указывает HTTP-метод для отправки формы. Это значение переопределяется атрибутом <code>formmethod</code> в элементах <code>&lt;button&gt;</code>, <code>&lt;input type="submit"&gt;</code> или <code>&lt;input type="image"&gt;</code>.</p> <p><b>Разрешенные значения:</b></p> <p><code>post</code> — данные формы включаются в тело HTTP-запроса. Метод является более надежным и безопасным, чем <code>get</code> и не имеет ограничений по размеру.</p> <p><code>get</code> — данные формы (пара имя-значение) добавляются в url-адрес с помощью разделителя <code>?</code> и отправляются на сервер. Данный способ имеет ограничения по размеру отправляемых данных и не подходит для отправки паролей и конфиденциальной информации.</p> <p><code>dialog</code> — используется, если отправка формы предназначена для закрытия диалогового окна, в котором находится форма.</p> <p><b>Синтаксис:</b> <code>method="post"</code></p>
name	<p>Задаёт имя формы, которое будет использоваться для доступа к элементам формы через сценарии. Значение должно быть уникальным среди элементов формы в коллекции форм, в которой оно находится, если таковые имеются.</p> <p><b>Синтаксис:</b> <code>name="login"</code></p>
novalidate	<p>Логический атрибут, который указывает, что форма не должна проверяться при отправке. Если этот атрибут не установлен, его можно переопределить атрибутом <code>formnovalidate</code> элемента <code>&lt;button&gt;</code>, <code>&lt;input type="submit"&gt;</code> или <code>&lt;input type="image"&gt;</code>, принадлежащих данной форме.</p> <p><b>Синтаксис:</b> <code>novalidate</code></p>
target	<p>Указывает, в каком окне выводить результат после отправки формы. Это значение переопределяется атрибутом <code>formtarget</code> в элементах <code>&lt;button&gt;</code>, <code>&lt;input type="submit"&gt;</code> или <code>&lt;input type="image"&gt;</code>.</p> <p><b>Разрешенные значения:</b></p> <p><code>_blank</code> — загружает ответ в новое окно/вкладку.</p> <p><code>_self</code> — загружает ответ в то же окно (значение по умолчанию).</p> <p><code>_parent</code> — загружает ответ в родительский фрейм. Если родителя нет, этот параметр ведет себя так же, как <code>_self</code>.</p> <p><code>_top</code> — загружает ответ во весь экран. Если родителя нет, этот параметр ведет себя так же, как <code>_self</code>.</p> <p><b>Синтаксис:</b> <code>target="_blank"</code></p>
rel	<p>Определяет связь между связанным ресурсом и текущим документом. <a href="#">Список разрешенных значений</a>.</p> <p><b>Синтаксис:</b> <code>rel="external"</code></p>

Элемент `<form>` представляет собой гиперссылку, которой можно управлять с помощью набора связанных элементов управления, значения которых можно отправить на сервер для обработки.

Атрибуты `action`, `enctype`, `method`, `novalidate` и `target` управляют поведением во время отправки формы.



```

<form method="post" enctype="application/x-www-form-urlencoded" action="https://pizza.example.com">
  <p><label>Customer name: <input name="custname" required autocomplete="shipping name"></label></p>
  <p><label>Telephone: <input type="tel" name="custtel" autocomplete="shipping tel"></label></p>
  <p><label>Buzzer code: <input name="custbuzz" inputmode="numeric"></label></p>
  <p><label>Email address: <input type="email" name="custemail" autocomplete="shipping email"></label></p>
  <fieldset>
    <legend>Pizza Size</legend>
    <p><label> <input type="radio" name="size" required value="small">Small</label></p>
    <p><label> <input type="radio" name="size" required value="medium">Medium</label></p>
    <p><label> <input type="radio" name="size" required value="large">Large</label></p>
  </fieldset>
  <fieldset>
    <legend>Pizza Toppings</legend>
    <p><label> <input type="checkbox" name="topping" value="bacon">Bacon</label></p>
    <p><label> <input type="checkbox" name="topping" value="cheese">Extra Cheese</label></p>
    <p><label> <input type="checkbox" name="topping" value="onion">Onion</label></p>
    <p><label> <input type="checkbox" name="topping" value="mushroom">Mushroom</label></p>
  </fieldset>
  <p><label>Preferred delivery time: <input type="time" min="11:00" max="21:00" step="900" name="delivery_time"></label></p>
  <p><label>Delivery instructions: <textarea name="comments" maxlength=1000></textarea></label></p>
  <p><button>Submit order</button></p>
</form>

```

## 2. Элемент <label>

**Категории контента:** потоковое содержимое, текстовое содержимое, интерактивное содержимое, видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибут `for`, который определяет, к какому полю формы привязан данный элемент. Значение атрибута должно содержать идентификатор поля формы.

Элемент `<label>` представляет надпись к элементу управления формы. Элемент может быть связан с конкретным полем формы с помощью атрибута `for`, либо путем помещения элемента управления формы внутрь `<label>`.

```

<p>
  <label for="username">Enter your username:</label>
  <input id="username">
</p>

```

```
<p>
  <label>Full name: <input name="fn"> <small>Format: First Last</small></label>
</p>
```

### 3. Элемент <input>

**Категории контента:** потоковое содержимое; текстовое содержимое; если не имеет атрибута `type="hidden"` — интерактивное содержимое; видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** отсутствует закрывающий тег.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 2. АТРИБУТЫ ЭЛЕМЕНТА <INPUT>

Атрибут	Описание, принимаемое значение
accept	<p>Строка, описывающая тип файла, который может быть выбран пользователем в элементе <code>&lt;input type="file"&gt;</code>. Допускается указывать расширение имени файла без учета регистра, начинающееся с символа <code>.</code>, например, <code>.jpg</code>, <code>.pdf</code> или <code>.doc</code> или MIME-типа без расширения:</p> <p><code>audio/*</code> — разрешает загрузку аудиофайлов;</p> <p><code>video/*</code> — разрешает загрузку видеофайлов;</p> <p><code>image/*</code> — разрешает загрузку изображений.</p> <p><b>Синтаксис:</b> <code>accept="image/*,.pdf"</code></p>
alt	<p>Отображает альтернативный текст для изображения, если оно отсутствует или не загружается по другой причине. Только для <code>&lt;input type="image"&gt;</code></p> <p><b>Синтаксис:</b> <code>alt="submit button"</code></p>
autocomplete	<p>Описывает, какие <a href="#">функции автозаполнения</a> должны предоставляться полям ввода. Типичная реализация автозаполнения вызывает предыдущие значения, введенные в то же поле ввода, но могут существовать и более сложные формы автозаполнения. Например, браузер может интегрироваться со списком контактов устройства для автозаполнения адресов электронной почты в поле ввода электронной почты.</p> <p>Атрибут <code>autocomplete</code> действителен для полей типа <code>hidden</code>, <code>text</code>, <code>search</code>, <code>url</code>, <code>tel</code>, <code>email</code>, <code>date</code>, <code>month</code>, <code>week</code>, <code>time</code>, <code>datetime-local</code>, <code>number</code>, <code>range</code>, <code>color</code> и <code>password</code>.</p> <p><b>Синтаксис:</b> <code>autocomplete="username"</code></p>
checked	<p>Логический атрибут, действителен для <code>&lt;input type="radio"&gt;</code> и <code>&lt;input type="checkbox"&gt;</code>. Если он присутствует в <code>type="checkbox"</code>, он указывает, что этот переключатель является выбранным в данный момент в группе одноименных переключателей. Если он присутствует в <code>type="radio"</code>, он указывает, что флажок установлен по умолчанию при загрузке страницы.</p>

	<p><b>Синтаксис:</b> checked</p>
dirname	<p>Разрешает отправку направления элемента и дает имя элемента управления, содержащего это значение, во время отправки формы. Устанавливается для <code>&lt;input type="text"&gt;</code> и <code>&lt;input type="search"&gt;</code> .</p> <p><b>Синтаксис:</b></p> <pre>&lt;form action="addcomment.cgi" method=post&gt; &lt;p&gt;&lt;label&gt;Comment: &lt;input type=text name="comment" dirname="comment.dir" required&gt;&lt;/label&gt;&lt;/p&gt; &lt;p&gt;&lt;button name="mode" type=submit value="add"&gt;Post Comment&lt;/button&gt;&lt;/p&gt; &lt;/form&gt;</pre> <p>Если пользователь введет в поле Hello , тело отправки может быть примерно таким:</p> <p>comment=Hello&amp;comment.dir=ltr&amp;mode=add , где направление элемента будет значением, установленным браузером.</p>
disabled	<p>Логический атрибут, который указывает, что пользователь не может редактировать и копировать содержимое поля. Отключенные поля ввода обычно отображаются более тусклым цветом или с использованием какой-либо другой формы индикации того, что поле недоступно для использования.</p> <p><b>Синтаксис:</b> disabled</p>
form	<p>Определяет форму, которой принадлежит данное поле. Значение должно соответствовать идентификатору элемента <code>&lt;form&gt;</code> в том же документе.</p> <p><b>Синтаксис:</b> form="formID"</p>
formaction	<p>Указывает url-адрес обработчика данных формы, отправляемых при нажатии на кнопку. Только для <code>&lt;input type="image"&gt;</code> и <code>&lt;input type="submit"&gt;</code> . Переопределяет значение атрибута action связанной формы.</p> <p><b>Синтаксис:</b> formaction="addcomment.cgi"</p>
formenctype	<p>Указывает, как данные должны кодироваться перед отправкой на сервер. Только для <code>&lt;input type="image"&gt;</code> и <code>&lt;input type="submit"&gt;</code> . Переопределяет значение атрибута enctype связанной формы.</p> <p><b>Разрешенные значения:</b></p> <p>application/x-www-form-urlencoded — значение по умолчанию. Данные формы кодируются как пары имя-значение, аналогично строке запроса URI. Это формат по умолчанию для POST. Тело HTTP-сообщения, отправляемого на сервер, представляет собой одну строку запроса — пары имя-значение разделены &amp; , а имена отделены от значений знаком =</p> <p>multipart/form-data — данные формы не кодируются. Это значение необходимо использовать для формы, содержащей элементы, управляющие загрузкой файлов. При отправке данных формы браузер разделяет каждый файл или вложение с помощью «границы из нескольких частей», которая является уникальным идентификатором, определяющим начало и конец каждой части. Границы части обозначаются линиями, начинающимися с тире. Это позволяет отправлять несколько частей в одном запросе и идентифицировать каждую из них со своими собственными метаданными, такими как MIME-тип, имя файла и т.д.</p> <p>text/plain — символы не кодируются, а пробелы заменяются на символ + . Полезен только для отладки.</p> <p><b>Синтаксис:</b> enctype="multipart/form-data"</p>

formmethod	<p>Указывает метод, который браузер будет использовать для отправки формы. Только для <code>&lt;input type="image"&gt;</code> и <code>&lt;input type="submit"&gt;</code>. Переопределяет значение атрибута <code>method</code> связанной формы.</p> <p><b>Синтаксис:</b> <code>formmethod="post"</code></p>
formnovalidate	<p>Логический атрибут, который указывает, что форма не подлежит проверке во время отправки. Только для <code>&lt;input type="image"&gt;</code> и <code>&lt;input type="submit"&gt;</code>. Переопределяет атрибут <code>novalidate</code> связанной формы.</p> <p><b>Синтаксис:</b> <code>formnovalidate</code></p>
formtarget	<p>Указывает, в каком окне выводить результат после отправки формы. Только для <code>&lt;input type="image"&gt;</code> и <code>&lt;input type="submit"&gt;</code>. Переопределяет значение атрибута <code>target</code> для связанной формы.</p> <p><b>Разрешенные значения:</b></p> <ul style="list-style-type: none"> <li><code>_blank</code> — загружает ответ в новое окно/вкладку.</li> <li><code>_self</code> — загружает ответ в то же окно (значение по умолчанию).</li> <li><code>_parent</code> — загружает ответ в родительский фрейм. Если родителя нет, этот параметр ведет себя так же, как <code>_self</code>.</li> <li><code>_top</code> — загружает ответ во весь экран. Если родителя нет, этот параметр ведет себя так же, как <code>_self</code>.</li> </ul> <p><b>Синтаксис:</b> <code>formtarget="_blank"</code></p>
height	<p>Только для <code>&lt;input type="image"&gt;</code>, задает высоту файла изображения, отображаемого в качестве графической кнопки отправки.</p> <p><b>Синтаксис:</b> <code>height="60"</code></p>
list	<p>Значение атрибута должно быть идентификатором элемента <code>&lt;datalist&gt;</code>, расположенного в том же документе. Для полей типа <code>text</code>, <code>search</code>, <code>url</code>, <code>tel</code>, <code>email</code>, <code>date</code>, <code>month</code>, <code>week</code>, <code>time</code>, <code>datetime-local</code>, <code>number</code>, <code>range</code> и <code>color</code>. Предоставляет список предопределенных значений, которые можно предложить пользователю для ввода.</p> <p><b>Синтаксис:</b> <code>list="datalistID"</code></p>
max	<p>Ограничивает ввод, определяя наибольшее значение в диапазоне допустимых значений. Значение может быть как целым, так и дробным числом. Для полей типа <code>date</code>, <code>month</code>, <code>week</code>, <code>time</code>, <code>datetime-local</code>, <code>number</code> и <code>range</code>.</p> <p>Если тип данных является периодическим (например, для дат или времени), значение <code>max</code> может быть меньше значения <code>min</code>, что указывает на то, что диапазон может повторяться; например, это позволяет указать временной диапазон от 22:00 до 4:00.</p> <p><b>Синтаксис:</b> <code>max="100"</code></p>
maxlength	<p>Определяет максимальное количество символов (в виде кодовых единиц UTF-16), которые пользователь может ввести в поле. Значение — целое число от 0 и выше. Это значение также должно быть больше или равно значению <code>minlength</code>.</p> <p><b>Синтаксис:</b> <code>maxlength="10"</code></p>
min	<p>Определяет самое отрицательное значение в диапазоне допустимых значений. Это значение должно быть меньше или равно значению атрибута <code>max</code>. Для полей <code>date</code>, <code>month</code>, <code>week</code>, <code>time</code>, <code>datetime-local</code>, <code>number</code> и <code>range</code>.</p>



	<b>Синтаксис:</b> <code>min="12"</code>
<code>minlength</code>	<p>Определяет минимальное количество символов, которое пользователь может ввести в поле ввода. Это должно быть неотрицательное целочисленное значение, меньшее или равное значению, заданному параметром <code>maxlength</code> .</p> <p><b>Синтаксис:</b> <code>minlength="24"</code></p>
<code>multiple</code>	<p>Логический атрибут, указывает, что пользователь может вводить адреса электронной почты, разделенные запятыми или может выбирать более одного файла при загрузке. Для полей типа <code>email</code> и <code>file</code> .</p> <p><b>Синтаксис:</b> <code>multiple</code></p>
<code>name</code>	<p>Значением атрибута должна быть строка, определяющая имя поля. Это имя передается вместе со значением при отправке данных формы. Если имя не указано или имя пусто, значение поля не отправляется вместе с формой. Не должно совпадать с именем типа поля, например, <code>name="checkbox"</code> .</p> <p><b>Синтаксис:</b> <code>name="username"</code></p>
<code>pattern</code>	<p>Значением атрибута выступает допустимое регулярное выражение JavaScript, которому должно соответствовать вводимое значение, чтобы это значение прошло проверку. При использовании атрибута <code>pattern</code> рекомендуется сообщать пользователю об ожидаемом формате, включив рядом поясняющий текст. Вы также можете включить атрибут <code>title</code> , большинство браузеров будут отображать этот заголовок как всплывающую подсказку. Для полей типа <code>tel</code> , <code>email</code> , <code>url</code> , <code>password</code> и <code>search</code> .</p> <p><b>Синтаксис:</b> <code>pattern="[a-z]{4,8}"</code></p>
<code>placeholder</code>	<p>Строка, которая дает пользователю краткую подсказку о том, какая информация ожидается в поле. Отображается в поле ввода до заполнения.</p> <p><b>Синтаксис:</b> <code>placeholder="Name"</code></p>
<code>readonly</code>	<p>Логический атрибут, который указывает, что пользователь не может изменять значение поля, выделение и копирование текста при этом доступно. Для полей типа <code>text</code> , <code>search</code> , <code>url</code> , <code>tel</code> , <code>email</code> , <code>date</code> , <code>month</code> , <code>week</code> , <code>time</code> , <code>datetime-local</code> , <code>number</code> и <code>password</code> .</p> <p><b>Синтаксис:</b> <code>readonly</code></p>
<code>required</code>	<p>Логический атрибут, который указывает, что данное поле является обязательным для заполнения. Если пользователь попытается отправить форму, не введя в это поле требуемое значение, то на экране отобразится предупреждающее сообщение.</p> <p><b>Синтаксис:</b> <code>required</code></p>
<code>size</code>	<p>Задаёт видимую ширину поля в символах. Значение по умолчанию — <code>20</code> . Для полей типа <code>email</code> , <code>password</code> , <code>tel</code> , <code>url</code> и <code>text</code> .</p> <p><b>Синтаксис:</b> <code>size="100"</code></p>
<code>src</code>	<p>Задаёт url-адрес изображения, используемого в качестве кнопки отправки данных формы. Только для поля <code>&lt;input type="image"&gt;</code> .</p> <p><b>Синтаксис:</b> <code>src="button.png"</code></p>
<code>step</code>	<p>Для элементов числовых типов ввода — <code>date</code> , <code>month</code> , <code>week</code> , <code>time</code> , <code>datetime-local</code> , <code>number</code> и <code>range</code> . Указывает величину увеличения или уменьшения значений в процессе регулировки диапазона (шаг).</p> <p><b>Синтаксис:</b> <code>step="2"</code></p>

type

Строка, определяющая тип отображаемого элемента управления.

**Разрешенные значения:**

hidden

Создает элемент управления, который не отображается, но значение которого отправляется на сервер.

text

Значение по умолчанию. Однострочное текстовое поле, разрывы строк автоматически удаляются из вводимого значения.

search

Однострочное текстовое поле для ввода строк поиска, разрывы строк автоматически удаляются из вводимого значения. Некоторые браузеры отображают иконку удаления — крестик, которую можно использовать для очистки поля.

tel

Поле для ввода номера телефона. Отображает клавиатуру телефона на некоторых устройствах с динамической клавиатурой.

url

Поле для ввода URL. Выглядит как поле для ввода текста, но имеет параметры проверки и соответствующую клавиатуру для поддержки браузеров и устройств с динамической клавиатурой.

email

Поле для ввода адреса электронной почты. Выглядит как ввод текста, но имеет параметры проверки и соответствующую клавиатуру для поддержки браузеров и устройств с динамической клавиатурой.

password

Однострочное текстовое поле, в котором вводимые пользователем символы заменяются на звездочки, маркеры, либо другие, установленные браузером значки. Предупредит пользователя, если сайт небезопасен.

date

ДД.ММ.ГГГГ

Элемент управления, открывает средство выбора даты (год, месяц и день, без времени).

month

-----

Элемент управления, открывает средство выбора месяца и года.

week

Неделя --, ----

Элемент управления, открывает средство выбора номера недели и года.

time

--:--

Элемент управления, позволяет вводить время в 24-часовом формате по шаблону чч:мм.


datetime-local

ДД.ММ.ГГГГ --:--

Элемент управления, позволяет вводить дату и время по шаблону дд.мм.гггг чч:мм.

number

Поле для ввода целочисленных значений. Атрибуты min, max и step задают верхний, нижний пределы и шаг между значениями соответственно. Эти атрибуты предполагаются у всех элементов, имеющих численные показатели, а их значения по умолчанию зависят от типа элемента.

range 

Элемент управления для ввода числа, точное значение которого не важно. Отображается как виджет диапазона со средним значением по умолчанию. Используется вместе с `min` и `max` для определения диапазона допустимых значений.

color 

Открывает виджет выбора цвета в шестнадцатеричном формате.

checkbox ☐

Отображает флажок, позволяющий выбирать/отменять выбор отдельных значений.

radio ☐

Отображает переключатель, позволяющий выбрать одно значение из нескольких вариантов с одинаковым значением `name`.

file  Файл не выбран

Поле для выбора одного или нескольких файлов из хранилища своего устройства. После выбора файлы можно загрузить на сервер с помощью отправки формы или манипулировать ими с помощью кода JavaScript и File API.

submit 

Отображает кнопку отправки формы.

image 

Создает графическую кнопку отправки формы, отображает изображение, определенное атрибутом `src` или значение атрибута `alt`, если изображение отсутствует.

reset 

Создает кнопку, которая сбрасывает содержимое формы к значениям по умолчанию.

button 

Создает кнопку без поведения по умолчанию, надписью к кнопке является значение атрибута `value`, по умолчанию пустое.

**Синтаксис:** `type="button"`

value

Строка, которая определяет текущее редактируемое значение для полей типа `text` и `password`; для полей типа `button`, `reset` и `submit` — текст на кнопке; для полей типа `checkbox`, `radio` и `hidden` — определяет связанное значение, которое отправляется на сервер.

**Синтаксис:** `value="medium"`

width

Только для `<input type="image">`, задает ширину файла изображения, отображаемого для представления графической кнопки отправки.

**Синтаксис:** `width="60"`

Элемент `<input>` представляет поле для ввода данных различных типов, часто отображаемое с виджетом, в зависимости от типа устройства и браузера.

HTML

```
<p><label>Preferred delivery time: <input type=time min="11:00" max="21:00" step="900"></p>
```

## 4. Элемент <button>

**Категории контента:** потоковое содержимое; текстовое содержимое; интерактивное содержимое; элемент, связанный с формой; видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 3. АТРИБУТЫ ЭЛЕМЕНТА <BUTTON>

Атрибут	Описание, принимаемое значение
disabled	Логический атрибут, отключает кнопку — ее нельзя нажать или сфокусировать.  <b>Синтаксис:</b> disabled
form	Указывает на форму, с которой связана кнопка. Значением этого атрибута является идентификатор элемента <form> в том же документе. По умолчанию кнопка связана со своим родительским элементом <form> , если он имеется.  <b>Синтаксис:</b> form="formID"
formaction	Указывает url-адрес обработчика данных формы, отправляемых при нажатии на кнопку. Только для кнопки типа submit . Переопределяет значение атрибута action , указанного для элемента <form> .  <b>Синтаксис:</b> formaction="subscribe.php"
formenctype	Только для кнопки типа submit . Указывает, как данные должны кодироваться перед отправкой на сервер. Переопределяет значение атрибута enctype связанной <form> .  <b>Разрешенные значения:</b> application/x-www-form-urlencoded — значение по умолчанию. Данные формы кодируются как пары имя-значение, аналогично строке запроса URI. Это формат по умолчанию для POST. Тело HTTP-сообщения, отправляемого на сервер, представляет собой одну строку запроса — пары имя-значение разделены & , а имена отделены от значений знаком =  multipart/form-data — данные формы не кодируются. Это значение необходимо использовать для формы, содержащей элементы, управляющие загрузкой файлов. При отправке данных формы браузер разделяет каждый файл или вложение с помощью «границы из нескольких частей», которая является уникальным идентификатором, определяющим начало и конец каждой части. Границы части обозначаются линиями, начинающимися с тире. Это позволяет отправлять несколько частей в одном запросе и идентифицировать каждую из них со своими собственными метаданными, такими как MIME-тип, имя файла и т.д.  text/plain — символы не кодируются, а пробелы заменяются на символ + . Полезен только для отладки.  <b>Синтаксис:</b> enctype="multipart/form-data"
formmethod	Атрибут определяет метод, который браузер будет использовать для отправки формы. Переопределяет значение атрибута method , указанного для элемента <form> . Указывается только для кнопок типа submit .

	<p><b>Разрешенные значения:</b></p> <p><code>get</code> — данные формы (пара имя-значение) добавляются в url-адрес с помощью разделителя <code>?</code> и отправляются на сервер. Данный способ имеет ограничения по размеру отправляемых данных и не подходит для отправки паролей и конфиденциальной информации.</p> <p><code>post</code> — данные формы включаются в тело HTTP-запроса. Метод является более надежным и безопасным, чем <code>get</code> и не имеет ограничений по размеру.</p> <p><code>dialog</code> — используется, если отправка формы предназначена для закрытия диалогового окна, в котором находится форма.</p> <p><b>Синтаксис:</b> <code>formmethod="post"</code></p>
<code>formnovalidate</code>	<p>Логический атрибут, указывает, что форма не подлежит проверке во время отправки. Переопределяет атрибут <code>novalidate</code> связанной формы. Указывается только для кнопок типа <code>submit</code>.</p> <p><b>Синтаксис:</b> <code>formnovalidate</code></p>
<code>formtarget</code>	<p>Атрибут задает, в каком окне выводить результат после отправки формы. Указывается только для кнопок типа <code>submit</code>. Переопределяет значение атрибута <code>target</code> для связанной формы.</p> <p><b>Разрешенные значения:</b></p> <p><code>_blank</code> — загружает ответ в новое окно/вкладку.</p> <p><code>_self</code> — загружает ответ в то же окно (значение по умолчанию).</p> <p><code>_parent</code> — загружает ответ в родительский фрейм. Если родителя нет, этот параметр ведет себя так же, как <code>_self</code>.</p> <p><code>_top</code> — загружает ответ во весь экран. Если родителя нет, этот параметр ведет себя так же, как <code>_self</code>.</p> <p><b>Синтаксис:</b> <code>formtarget="_blank"</code></p>
<code>name</code>	<p>Определяет имя кнопки, используемое для отправки формы. На сервер отправляется пара <code>имя=значение</code>, где имя задается атрибутом <code>name</code>, а значение — атрибутом <code>value</code>. Значение может совпадать с текстом на кнопке, а может отличаться.</p> <p><b>Синтаксис:</b> <code>name="Cancel"</code></p>
<code>type</code>	<p>Управляет поведением кнопки при ее активации.</p> <p><b>Разрешенные значения:</b></p> <p><code>submit</code> — отправляет данные формы, значение по умолчанию.</p> <p><code>reset</code> — сбрасывает данные формы в их исходные значения.</p> <p><code>button</code> — кнопка не имеет поведения по умолчанию и ничего не делает при нажатии по умолчанию. Если кнопка на странице не предназначена для отправки данных формы на сервер, установка этого значения обязательна.</p> <p><b>Синтаксис:</b> <code>type="submit"</code></p>
<code>value</code>	<p>Определяет значение, связанное с именем кнопки, когда она отправляется вместе с данными формы. Кнопка (и ее значение) включается в отправку формы только в том случае, если сама кнопка использовалась для инициирования отправки формы. Также атрибут <code>value</code> используется для доступа к данным через JavaScript.</p> <p><b>Синтаксис:</b> <code>value="Cancel"</code></p>

Элемент `<button>` представляет собой кнопку, помеченную своим содержимым. Атрибут `type` управляет поведением кнопки при ее активации.

HTML

```
<button type="button" onclick="alert('This 15-20 minute piece was composed by George Gershwin.
```

5. Элемент `<select>`

**Категории контента:** потоковое содержимое; текстовое содержимое; интерактивное содержимое; элемент, связанный с формой; видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 4. АТРИБУТЫ ЭЛЕМЕНТА `<SELECT>`

Атрибут	Описание, принимаемое значение
autocomplete	Значение атрибута содержит строку, которая предоставляет подсказку для <a href="#">функции автозаполнения</a> браузера.  <b>Синтаксис:</b> autocomplete="bday"
disabled	Логический атрибут, который указывает, что пользователь не может взаимодействовать с элементом.  <b>Синтаксис:</b> disabled
form	Указывает на форму, с которой связан элемент. Значением атрибута должен быть идентификатор формы в том же документе.  <b>Синтаксис:</b> form="formID"
multiple	Логический атрибут, который указывает, что в списке можно выбрать несколько параметров. Большинство браузеров будут отображать окно списка с прокруткой вместо выпадающего списка с одной строкой.  <b>Синтаксис:</b> multiple
name	Определяет имя для выпадающего списка, отражающее его тематику.  <b>Синтаксис:</b> name="pets"
required	Логический атрибут. Выводит сообщение о том, что пользователь должен выбрать значение из раскрывающегося списка перед отправкой формы.  <b>Синтаксис:</b> required
size	Если для элемента <code>&lt;select&gt;</code> указан атрибут <code>multiple</code> , этот атрибут представляет количество строк в списке, которые должны быть видны одновременно. Значение по умолчанию — 0.

Элемент `<select>` представляет собой элемент управления для выбора среди набора параметров.

HTML

```
<select name="unittype" required>
  <option value="">Select unit type</option>
  <option value="1">Miner</option>
  <option value="2">Puffer</option>
  <option value="3">Snipey</option>
  <option value="4">Max</option>
  <option value="5">Firebot</option>
</select>
```

## 6. Элемент `<datalist>`

Поддержка браузерами

“

IE: 10 ЧП  
 Edge: 79  
 Firefox: 4 ЧП  
 Chrome: 69  
 Safari: 12.1  
 Opera: 64  
 iOS Safari: 12.2  
 Opera Mini: —  
 Android Browser: 4.4.4  
 Chrome for Android: 97

**Категории контента:** потоковое содержимое, текстовое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#).

Элемент `<datalist>` содержит набор элементов `<option>`, которые представляют допустимые или рекомендуемые значения, доступные для выбора в элементе `<input>`. Эти значения отображаются в `<input>` в виде раскрывающегося списка и отфильтровываются по мере ввода данных в поле.

При рендеринге страницы элемент `<datalist>` ничего не отображает, он скрыт вместе со своими дочерними элементами.

Элемент `<datalist>` подключается к элементу `<input>` с помощью атрибута `list`, при этом значение `id` элемента `<datalist>` должно совпадать со значением атрибута `list` элемента `<input>`.



```
<label>
  Animal:
  <input name="animal" list="animals">
  <datalist id="animals">
    <option value="Cat">
    <option value="Dog">
  </datalist>
</label>
```

## 7. Элемент <optgroup>

**Категории контента:** отсутствует.

**Контекст, в котором этот элемент может быть использован:** как дочерний элемент элемента <select> .

**Пропуск тегов:** закрывающий тег элемента <optgroup> может быть опущен, если за элементом <optgroup> сразу следует другой элемент <optgroup> или если в родительском элементе больше нет содержимого.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 5. АТРИБУТЫ ЭЛЕМЕНТА <OPTGROUP>

Атрибут	Описание, принимаемое значение
disabled	<p>Логический атрибут, отключает группу элементов &lt;option&gt; . Часто браузеры выделяют такой элемент управления серым цветом, и он не получает никаких событий просмотра, таких как щелчки мыши или связанные с фокусом.</p> <p><b>Синтаксис:</b> disabled</p>
label	<p>Задаёт имя группы параметров. Не является обязательным.</p> <p><b>Синтаксис:</b> label="Group 1"</p>

Элемент <optgroup> представляет собой группу элементов <option> с общей надписью, обычно выделенной жирным шрифтом. Браузеры отображают такие группы как связанные друг с другом, отдельно от других элементов <option> .

```
<form action="courseselector.dll" method="get">
  <p>Which course would you like to watch today?
  <p><label>Course:
    <select name="c">
      <optgroup label="8.01 Physics I: Classical Mechanics">
        <option value="8.01.1">Lecture 01: Powers of Ten
        <option value="8.01.2">Lecture 02: 1D Kinematics
        <option value="8.01.3">Lecture 03: Vectors
      <optgroup label="8.02 Electricity and Magnetism">
        <option value="8.02.1">Lecture 01: What holds our world together?
```



```

    <option value="8.02.2">Lecture 02: Electric Field
    <option value="8.02.3">Lecture 03: Electric Flux
  <optgroup label="8.03 Physics III: Vibrations and Waves">
    <option value="8.03.1">Lecture 01: Periodic Phenomenon
    <option value="8.03.2">Lecture 02: Beats
    <option value="8.03.3">Lecture 03: Forced Oscillations with Damping
  </optgroup>
</select>
</label>
<p><input type=submit value="▶ Play">
</form>

```

## 8. Элемент <option>

**Категории контента:** отсутствует.

**Контекст, в котором этот элемент может быть использован:** как дочерний элемент элемента <select>, <datalist> или <optgroup>.

**Пропуск тегов:** закрывающий тег элемента <option> может быть опущен, если за элементом <option> сразу же следует другой элемент <option>, или если за ним сразу следует элемент <optgroup>, или если в родительском элементе больше нет содержимого.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 6. АТРИБУТЫ ЭЛЕМЕНТА <OPTION>

Атрибут	Описание, принимаемое значение
disabled	<p>Логический атрибут, если установлен, элемент не получает никаких событий просмотра, таких как щелчки мыши или связанные с фокусом.</p> <p><b>Синтаксис:</b> disabled</p>
label	<p>Представляет текст для надписи, указывающий значение параметра. Если атрибут label не определен, его значение равно текстовому содержимому элемента.</p> <p><b>Синтаксис:</b> label="Испания"</p>
selected	<p>Логический атрибут, помечает данный вариант как выбранный изначально.</p> <p><b>Синтаксис:</b> selected</p>
value	<p>Содержимое атрибута представляет собой значение, которое будет отправлено с формой, если выбран этот вариант. Если опущен, значение берется из текстового содержимого элемента.</p> <p><b>Синтаксис:</b> value="120 м"</p>

Элемент <option> представляет собой вариант в элементе <select> или часть списка в элементе <datalist>. В некоторых случаях элемент <option> может выступать как надпись-заполнитель в элементе <select>.

## 9. Элемент <textarea>



**Категории контента:** потоковое содержимое; текстовое содержимое; интерактивное содержимое; элемент, связанный с формой; видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 7. АТРИБУТЫ ЭЛЕМЕНТА <TEXTAREA>

Атрибут	Описание, принимаемое значение
autocomplete	<p>Указывает, может ли значение элемента автоматически заполняться браузером.</p> <p><b>Разрешенные значения:</b></p> <p>off — пользователь должен явно вводить значение в это поле для каждого использования.</p> <p>on — браузер может автоматически дополнять значение на основе значений, которые пользователь ввел во время предыдущих использований.</p> <p><b>Синтаксис:</b> autocomplete="off"</p>
cols	<p>Устанавливает ширину текстовой области через количество символов. Когда пользователь вводит больше текста, появляется полоса прокрутки. Если не указано, значение по умолчанию равно 20 .</p> <p><b>Синтаксис:</b> cols="100"</p>
dirname	<p>Разрешает отправку направления элемента и дает имя элемента управления, содержащего это значение, во время отправки формы.</p> <p><b>Синтаксис:</b> dirname="comment.dir"</p>
disabled	<p>Логический атрибут, отключает возможность редактирования и копирования содержимого поля.</p> <p><b>Синтаксис:</b> disabled</p>
form	<p>Указывает на форму, с которой связан элемент. Значением этого атрибута является идентификатор формы в том же документе.</p> <p><b>Синтаксис:</b> form="formID"</p>
maxlength	<p>Указывает максимальное количество символов (кодowych единиц UTF-16), которое может ввести пользователь.</p> <p><b>Синтаксис:</b> maxlength="200"</p>
minlength	<p>Указывает минимальное количество символов, которое должен ввести пользователь.</p> <p><b>Синтаксис:</b> minlength="10"</p>
name	<p>Задаёт имя текстового поля, используемое для отправки формы.</p> <p><b>Синтаксис:</b> name="comment"</p>
placeholder	<p>Выводит подсказку для пользователя, что можно ввести в данное поле. Возврат каретки или перевод строки в тексте-заполнителе отображаются как разрывы строк при отображении подсказки. Следует использовать только для демонстрации примера</p>

	<p>типа данных, которые следует вводить в форму; они не заменяют элемент <code>&lt;label&gt;</code>, привязанный к текстовому полю.</p> <p><b>Синтаксис:</b> <code>placeholder="Enter your comment here..."</code></p>
readonly	<p>Логический атрибут, который указывает, что пользователь не может изменить текстовое значение поля. В отличие от атрибута <code>disabled</code>, не запрещает пользователю щелкать или копировать текст.</p> <p><b>Синтаксис:</b> <code>readonly</code></p>
required	<p>Логический атрибут, выводит сообщение о том, что данное поле является обязательным для заполнения.</p> <p><b>Синтаксис:</b> <code>required</code></p>
rows	<p>Задаёт количество видимых строк текста для элемента.</p> <p><b>Синтаксис:</b> <code>rows="10"</code></p>
wrap	<p>Указывает, как переносится текст в элементе.</p> <p><b>Разрешенные значения:</b></p> <p><code>hard</code> — браузер автоматически вставляет разрывы строк, чтобы ширина каждой строки не превышала ширину элемента; атрибут <code>cols</code> также должен быть указан, чтобы значение применилось.</p> <p><code>soft</code> — значение по умолчанию, браузер не вставляет никаких дополнительных разрывов строк.</p> <p><code>off</code> — браузер не вставляет никаких дополнительных разрывов строк, а сегменты строк, превышающие <code>cols</code>, не переносятся, и элемент становится прокручиваемым по горизонтали.</p> <p><b>Синтаксис:</b> <code>wrap="hard"</code></p>

Элемент `<textarea>` представляет собой многострочный элемент управления для ввода обычного текста.

HTML

```
<p>If you have any comments, please let us know: <textarea cols="80" name="comments" maxlength=
```

## 10. Элемент `<output>`

**Категории контента:** потоковое содержимое; текстовое содержимое; элемент, связанный с формой; видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

Атрибут	Описание, принимаемое значение
for	Указывает разделенный пробелами список идентификаторов элементов с входными данными, которые участвовали в расчете. <b>Синтаксис:</b> for="a b"
form	Указывает идентификатор связанной формы. <b>Синтаксис:</b> form="formID"
name	Определяет имя, на которое можно будет ссылаться из обработчиков событий элементов управления формы; само значение элемента не передается при отправке формы. <b>Синтаксис:</b> name="outputName"

Элемент `<output>` представляет элемент-контейнер, в который сайт или приложение может выводить результаты вычислений или результат действия пользователя.

HTML

```
<form onsubmit="return false" oninput="o.value = a.valueAsNumber + b.valueAsNumber">
  <input id="a" type="number" step="any"> +
  <input id="b" type="number" step="any"> =
  <output id="o" for="a b"></output>
</form>
```

## 11. Элемент `<progress>`

**Категории контента:** потоковое содержимое, текстовое содержимое, маркируемый элемент, видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибут `value`, который задает текущее значение элемента, и атрибут `max`, который задает верхнюю границу диапазона. Если не указано максимальное значение, по умолчанию оно равно 1.



Элемент `<progress>` выводит на экран виджет, который отображает индикатор выполнения задачи. С его помощью пользователи могут отслеживать ход выполнения длительной операции. Индикатор выполнения может либо показать приблизительный процент завершения, либо указать, что операция выполняется.

HTML

```
<section>
  <h2>Task Progress</h2>
  <p>Progress: <progress id="p" max="100"><span>0</span>%</progress></p>
```

```

<script>
    var progressBar = document.getElementById("p");
    function updateProgress(newValue) {
        progressBar.value = newValue;
        progressBar.getElementsByTagName("span")[0].textContent = newValue;
    }
</script>
</section>

```

## 12. Элемент <meter>

**Категории контента:** потоковое содержимое, текстовое содержимое, маркируемый элемент, видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 9. АТРИБУТЫ ЭЛЕМЕНТА <METER>

Атрибут	Описание, принимаемое значение
value	<p>Задаёт текущее числовое значение. Оно должно быть между минимальным и максимальным значениями, если они указаны. Если не указано или имеет неправильный формат, значение равно 0. Если указано, но не в пределах заданного диапазона, значение равно верхней границе диапазона.</p> <p><b>Синтаксис:</b> value="0.75"</p>
min	<p>Задаёт нижнюю границу диапазона. Значение по умолчанию 0.</p> <p><b>Синтаксис:</b> min="10"</p>
max	<p>Задаёт верхнюю границу диапазона. Значение по умолчанию 1.</p> <p><b>Синтаксис:</b> max="75"</p>
low	<p>Задаёт верхнюю числовую границу нижнего предела диапазона. Если не указано или меньше минимального, то равно минимальному значению.</p> <p><b>Синтаксис:</b> low="7"</p>
high	<p>Задаёт нижнюю числовую границу верхнего предела диапазона. Если не указано или больше максимального значения, то равно максимальному значению.</p> <p><b>Синтаксис:</b> high="10"</p>
optimum	<p>Указывает оптимальное числовое значение в пределах диапазона. При использовании с атрибутами low и high указывает, что предпочтительным считается более высокий диапазон. Если значение находится между атрибутами min и low, предпочтительным считается более низкий диапазон. Браузер может по-разному раскрасить полосу счетчика в зависимости от того, меньше ли значение оптимального значения или равно ему.</p> <p><b>Синтаксис:</b> optimum="8"</p>



Элемент `<meter>` представляет собой скалярное измерение в пределах известного диапазона, либо дробное значение; например, использование дискового пространства, релевантность результата запроса или доля голосующего населения, выбравшего конкретного кандидата.

HTML

```
<p>The grapefruit pie had a radius of 12cm and a height of
2cm.</p>
<dl>
  <dt>Radius:
  <dd> <meter min="0" max="20" value="12">12cm</meter>
  <dt>Height:
  <dd> <meter min="0" max="10" value="2">2cm</meter>
</dl>
```

13. Элемент `<fieldset>`

**Категории контента:** потоковое содержимое; корневое секционное содержимое; элемент, связанный с формой; видимое содержимое.

**Контекст, в котором этот элемент может быть использован:** где ожидается текстовое содержимое.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#), а также атрибуты, приведенные в таблице ниже:

ТАБЛИЦА 10. АТРИБУТЫ ЭЛЕМЕНТА `<FIELDSET>`

Атрибут	Описание, принимаемое значение
disabled	Логический атрибут, отключает от редактирования и взаимодействия все дочерние элементы управления формой, за исключением находящихся внутри элемента <code>&lt;legend&gt;</code>  Синтаксис: disabled
form	Указывает на связанную форму. Значением атрибута является id элемента <code>&lt;form&gt;</code> в том же документе.  Синтаксис: form="formId"
name	Определяет имя, которое будет использоваться для доступа к элементу с помощью JavaScript, или для ссылки на данные формы после заполнения и отправки формы. Является аналогом атрибута id .  Синтаксис: name="fieldsetName"

Элемент `<fieldset>` группирует элементы управления формой и надписи `<label>` к ним, не обязательно с заголовком.



```
<fieldset name="clubfields" disabled>
  <legend>
    <label>
      <input type="checkbox" name="club" onchange="form.clubfields.disabled = !checked">
      Use Club Card
    </label>
  </legend>
  <p><label>Name on card: <input name="clubname" required></label></p>
  <p><label>Card number: <input name="clubnum" required pattern="[-0-9]+"></label></p>
  <p><label>Expiry date: <input name="clubexp" type="month"></label></p>
</fieldset>
```

## 14. Элемент <legend>

**Категории контента:** отсутствуют.

**Контекст, в котором этот элемент может быть использован:** как первый дочерний элемент элемента <fieldset>.

**Пропуск тегов:** ни один из тегов не может быть пропущен.

Для элемента доступны [глобальные атрибуты](#).

Элемент <legend> представляет собой заголовок для остального содержимого родительского элемента <fieldset>.

## 15. Атрибут autocomplete

Атрибут autocomplete позволяет указывать, какие разрешения должен иметь браузер, чтобы предоставлять автоматическую помощь в заполнении значений полей формы, а также давать указания относительно типа информации, ожидаемой в поле.

Атрибут доступен для элемента <input>, который принимает текстовое или числовое значение в качестве входных данных, элементов <textarea>, <select> и <form>.

Источник предлагаемых значений зависит от браузера; обычно значения берутся из прошлых значений, введенных пользователем, но они также могут поступать из предварительно сконфигурированных значений. Например, браузер может позволить пользователю сохранять свое имя, адрес, номер телефона и адреса электронной почты для целей автозаполнения. Возможно, браузер предлагает возможность сохранять зашифрованную информацию о кредитной карте для автозаполнения после процедуры аутентификации.

### Разрешенные значения

off — отключает автозаполнение, пользователь должен каждый раз вводить значения каждого поля.

on — включает автозаполнение, при этом никаких указаний относительно типа данных, ожидаемых в поле, для браузера не предоставляется.



`name` — полное имя человека. Предпочтительнее использовать данное значение, а не разбивать имя на его компоненты, чтобы избежать большого разнообразия человеческих имен. Если нужно разбить имя на компоненты, можно использовать следующие значения: `honorific-prefix`, `given-name`, `additional-name`, `family-name`, `honorific-suffix`.

`nickname` — псевдоним или сокращенное имя.

`organization-title` — должность, которую человек имеет в организации, например «Инженер-программист», «Старший вице-президент», «Заместитель управляющего директора».

`username` — имя пользователя или имя учетной записи.

`new-password` — новый пароль. При создании новой учетной записи или изменении паролей это следует использовать для поля «Введите новый пароль» или «Подтвердите новый пароль», а не для общего поля «Введите текущий пароль», которое может присутствовать. Может использоваться браузером как для предотвращения случайного ввода существующего пароля, так и для предложения помощи в создании безопасного пароля.

`current-password` — текущий пароль пользователя.

`one-time-code` — одноразовый код, используемый для проверки личности пользователя.

`organization` — название компании, соответствующее лицу, адресу или контактной информации в других полях, связанных с этим полем.

`street-address` — адрес, который содержит название улицы, номер дома и квартиру. Это может быть несколько строк текста, и он должен полностью идентифицировать местоположение адреса на втором административном уровне (обычно это город), но не должен включать название города, почтовый индекс или название страны. `address-line1`, `address-line2` и `address-line3` представляют каждую отдельную строку адреса. Они должны присутствовать только в том случае, если `street-address` отсутствует.

`address-level4`, `address-level3`, `address-level2` и `address-level1` — используются для описания местонахождения почтового адреса. Разные страны имеют разное количество уровней. Например, в США используется два уровня (штат и город), в Великобритании — один или два в зависимости от адреса (почтовый город и, в некоторых случаях, населенный пункт), а в Китае — три (провинция, город, район). Поле `address-level1` представляет самую широкую административную единицу. Различные страны упорядочивают поля по-разному; например, в США город (уровень 2) предшествует штату (уровень 1); в то время как в Японии префектура (уровень 1) предшествует городу (уровень 2), который предшествует району (уровень 3). Авторам рекомендуется предоставлять формы, которые представлены в соответствии с принятыми в стране правилами (скрытие, отображение и перестановка полей соответственно по мере того, как пользователь меняет страну).

`country` — код страны в соответствии с [ISO 3166](#).

`country-name` — название страны, в некоторых случаях производное от `country`.

`postal-code` — почтовый индекс, CEDEX-код.

`cc-name` — полное имя, напечатанное или связанное с платежным средством, таким как кредитная карта. Обычно предпочтительнее использовать поле `cc-name`, чем разбивать имя на части, такие как `cc-given-name`, `cc-additional-name` и `cc-family-name`.



`cc-number` — номер кредитной карты или другой номер, идентифицирующий способ оплаты, например номер счета.

`cc-exp` — срок действия платежного инструмента, например, 2021-12. Также можно разбить на отдельные параметры — `cc-exp-month` и `cc-exp-year`.

`cc-csc` — код безопасности для платежного инструмента (также известный как код безопасности карты CSC, код проверки карты CVC, значение проверки карты CVV, код панели подписи SPC, идентификатор кредитной карты CCID и т.д.).

`cc-type` — тип платежного инструмента, например, Visa.

`transaction-currency` — валюта, в которой должна быть совершена транзакция, в формате [ISO 4217](#).

`transaction-amount` — сумма транзакции в указанной валюте.

`language` — предпочтительный язык, указанный в действующем языковом коде [BCP 47](#).

`bday` — полная дата рождения, например, 1955-06-08. Можно разбить на отдельные компоненты: `bday-day` — целое число в диапазоне от 1 до 31, `bday-month` — целое число в диапазоне от 1 до 12, `bday-year`.

`sex` — гендерная идентичность в виде текста произвольной формы без новой строки.

`url` — домашняя страница или другая веб-страница, соответствующая компании, лицу, адресу или контактной информации в других полях, связанных с этим полем.

`photo` — фотография, иконка или другое изображение, соответствующее компании, лицу, адресу или контактной информации в других полях, связанных с этим полем.

`tel` — полный номер телефона, включая код страны. Если нужно разбить номер телефона на его компоненты, вы можете использовать эти значения для этих полей: `tel-country-code`, `tel-national`, `tel-area-code`, `tel-local`, `tel-local-prefix`, `tel-local-suffix`.

`tel-extension` — добавочный или внутренний номер абонента.

`email` — адрес электронной почты.

`impp` — URL-адрес, представляющий конечную точку протокола обмена мгновенными сообщениями, например, `aim:goim?screenname=example` или `xmpp:fred@example.net`.

По материалам [Forms](#)

---

Табличные данные

Интерактивные элементы

---

Поделиться:

