

Формы в HTML – это самая сложная, но с другой стороны, пожалуй самая интересная тема в HTML.

Веб-формы позволяют посетителям сайта внести в специальные поля ту или иную информацию, а разработчику ее получить в удобном для него виде.

Примером формы может служить гостевая книга, анкета, онлайн тест. Формы заполняют при регистрации на сайте, при оформлении заказов в интернет магазине и т.п.

Пользуясь HTML, Вы можете создать каркас формы: текстовые поля, меню, списки, кнопки, флажки и переключатели. То есть те элементы, при помощи которых в форму вносится определенная информация.

Затем данные, внесенные в форму, отправляются на сервер для обработки. Но HTML здесь бессилён – над обработкой формы уже работает программа или скрипт, который к ней привязывается. Такие программы обычно пишут на языке php или javascript.

Атрибуты формы – тег `<form>`

Форм на веб-странице может несколько (*столько, сколько нужно разработчику*). Каждая из них начинается тегом `<form>` и завершается закрывающим тегом `</form>`.

Атрибут `action` является обязательным для любой формы – он указывает адрес к файлу, который обслуживает форму (*обрабатывает внесенные в нее данные*).

Атрибут `method` определяет способ отправки содержимого формы. Существует два метода – `GET` и `POST`. Сейчас не имеет смысла вникать в эти параметры, так как тема отправки информации методами `GET` и `POST` относится к языкам обработки данных (*например, PHP*). Достаточно знать, что именно метод передачи данных `POST` в большинстве случаев используется в формах.

Атрибут `name` тега `<form>` не является обязательным. Но если в документе несколько форм, то каждая из них должна быть как-то идентифицирована обработчиком. Поэтому наличие атрибута `name` в этом случае необходимо – он задает уникально имя формы.

Можно также установить кодировку для вводимых данных - за это отвечает атрибут `accept-charset`, а также, при помощи атрибута `target`, определить окно, в котором будет отображаться результат обработки отправленной формы (*в новом или в текущем окне*).

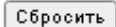
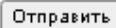
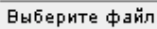

Но сам по себе тег `<form>` не имеет смысла, ведь форма передает данные, которые сначала нужно куда-то внести!

Ввод данных. Поля формы - тег `<input>`

Тег `<input>` это наиболее часто встречающийся в формах тег. Он предназначен для создания различных элементов, служащих для ввода данных в форму: это текстовые поля, кнопки, флажки, переключатели.

`type` - это основной атрибут тега `<input>`. Он устанавливает тип поля (*элемента*) формы:

Значение атрибута <code>type="..."</code>	Результат	Описание
<code><input type="text" size="20" value="Ваше Имя..."></code>	<input type="text" value="Ваше Имя..."/>	Однострочное текстовое поле для ввода текста. Атрибут <code>size</code> задает ширину поля в символах.
<code><input type="password" maxlength="10" size="20"></code>	<input type="password"/>	Текстовое поле для ввода пароля. Атрибут <code>maxlength</code> устанавливает максимальное количество символов, которое можно ввести
<code><input type="radio" name="radio" value="1" checked></code> <code><input type="radio" name="radio" value="2"></code> <code><input type="radio" name="radio" value="3"></code>	<input checked="" type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3	Переключатель. Возможен выбор лишь одного варианта из предложенных. Атрибут <code>checked</code> определяет заранее помеченное поле.
<code><input type="button" value="Кнопка"></code>	<input type="button" value="Кнопка"/>	Кнопка. Атрибут <code>value</code> устанавливает надпись на кнопке.

<code><input type="reset" value="Сбросить"></code>		Кнопка сброса. Возвращает поля формы к их первоначальному виду. Сброс внесенных данных.
<code><input type="submit"></code>		Кнопка для отправки внесенных данных.
<code><input type="file"></code>	 файл не выбран	Поле для ввода имени пересылаемого файла.
<code><input type="image" src="img/button.gif"></code>		Кнопка-изображение. Служит также для отправки данных на сервер. Атрибут <code>src</code> указывает адрес файла с изображением.
<code><input type="hidden" size="20"></code>		Скрытое поле - невидимое для пользователя.

Раскрывающийся список – теги `<select>` и `<option>`

Тег `<select>` также как и тег `<input>` служит для сбора информации – он создает список, из которого возможен выбор одного или нескольких элементов. Каждому элементу соответствует какое-либо значение, которое и отправляется на сервер для обработки.

Вид создаваемого списка зависит от значения атрибута `size`: при `size="1"` (значение по умолчанию) список будет раскрывающимся.

Иное значение атрибута `size` будет соответствовать количеству отображаемых пунктов списка. Например, при `size="3"`, видимыми будут три элемента. Для просмотра остальных пунктов списка (если таковые есть) следует воспользоваться вертикальной полосой прокрутки, которая добавляется автоматически.

По умолчанию возможен выбор лишь одного элемента списка. Добавление атрибута `multiple` к тегу `<select>` позволяет выбрать несколько пунктов списка с помощью мыши и клавиш `Ctrl` и `Shift`.

При наличии атрибута `multiple`, будут отображены все пункты списка или их часть (в зависимости от браузера), если атрибутом `size` не установлено определенное количество видимых пунктов.

Обязательным элементом раскрывающегося списка является тег `<option>`. Он располагается внутри тега `<select>` и создает каждый пункт списка.

При помощи атрибута `name` тега `<select>` списку дается имя, а при помощи атрибута `value` тега `<option>` элементу списка присваивают значение. Таким образом на сервер отправляется пара «имя=значение» - имя списка и значение выбранного элемента/ов.

Атрибут `selected` тега `<option>` определяет заранее выделенный пункт списка, а атрибут `disabled` блокирует элемент списка - его нельзя выбрать.

Фрагмент кода

```
<form>
<p> Сторона света - одно из четырех основных направлений: </p>
<select name="storona">
<option value="North"> Север </option>
<option value="South"> Юг </option>
<option value="West"> Запад </option>
<option value="East"> Восток </option>
</select>
<p> 7 Чудес света! </p>
<select name="chudo" size="4" multiple>
<option value="1"> Пирамида Хеопса </option>
<option value="2" selected> Висячие сады Семирамиды </option>
<option value="3" disabled> Статуя Зевса в Олимпии </option>
<option value="4"> Храм Артемиды в Эфесе </option>
<option value="5"> Мавзолей в Галикарнасе </option>
<option value="6"> Колосс Родосский </option>
<option value="7"> Александрийский маяк </option>
</select>
```

```
<p> <input type="reset" value="Сброс"> </p>  
</form>
```

Результат

Сторона света - одно из четырех основных направлений:

Север ▾

7 Чудес света!

Пирамида Хеопса
Висячие сады Семирамиды
Статуя Зевса в Олимпии
Храм Артемиды в Эфесе

Сброс

Многострочное текстовое поле - тег <textarea>

Тег <textarea> создает многострочное текстовое поле. В нем допустимо делать переносы строк, которые сохраняются при передаче данных обработчику.

Атрибут `name` тега <textarea> является обязательным - при помощи его текстовому полю присваивают имя.

Атрибут `disabled` блокирует поле - содержимое поля нельзя изменить и оно недоступно. Атрибут `readonly` говорит о том, что поле предназначено только для чтения - пользователь не имеет возможность редактировать содержимое, но оно доступно - его можно выделить и, например, скопировать.

Можно задать ширину текстового поля в символах и высоту поля в строках при помощи атрибутов `cols` и `rows` соответственно.

Фрагмент кода

```
<p> <textarea name="text" cols="40" rows="5" readonly> Если содержимое поля превысит его  
размеры - появится бегунок. </textarea> </p>
```