**Тег <form>** устанавливает форму на веб-странице. Форма предназначена для обмена данными между пользователем и сервером. Область применения форм не ограничена отправкой данных на сервер, с помощью клиентских скриптов можно получить доступ к любому элементу формы, изменять его и применять по своему усмотрению.

Документ может содержать любое количество форм, но одновременно на сервер может быть отправлена только одна форма. По этой причине данные форм должны быть независимы друг от друга.

У формы есть следующие атрибуты:

* **action**-ссылка на файл,который будет обработать данные формы.
* **method**–указывает на метод запроса к серверу. Может принимать два параметра: GETиPOST. По умолчанию GET.
* **target** –указывает,как будут отображены результаты, в текущем окне или в новой вкладке.
* **autocomplete** -контролирует автозаполнение форм, имеет проблемы с кроссбраузерностью.

Для отправки формы на сервер используется кнопка Submit, того же можно добиться, если нажать клавишу Enter в пределах формы. Если кнопка Submit отсутствует в форме, клавиша Enter имитирует ее использование.

Когда форма отправляется на сервер, управление данными передается программе, заданной атрибутом action тега <form>. Предварительно браузер подготавливает информацию в виде пары «имя=значение», где имя определяется атрибутом name тега <input>, а значение введено пользователем или установлено в поле формы по умолчанию. Если для отправки данных используется **метод GET**, то адресная строка может принимать следующий вид.

<http://www.htmlbook.ru/cgi-bin/handler.cgi?nick=%C2%E0%ED%FF+%D8%E0%EF%EE%F7%EA%E8%ED&page=5>

Параметры перечисляются после вопросительного знака, указанного после адреса CGI-программы и разделяются между собой символом амперсанда (&). Нелатинские символы преобразуются в шестнадцатеричное представление (в форме %HH, где HH — шестнадцатеричный код для значения ASCII-символа), пробел заменяется на плюс (+).

Допускается внутрь контейнера <form> помещать другие теги, при этом сама форма никак не отображается на веб-странице, видны только ее элементы и результаты вложенных тегов.

**POST**–позволяет отправлять большее количество данных, чем доступно методу GET, поскольку у последнего установлено ограничение в 4 Кб.

Большие объемы данных используются в форумах, почтовых службах, при заполнении базы данных, при пересылке файлов и пр.

Тег <input>

Тег <input> является одним из разносторонних элементов формы и позволяет создавать разные элементы интерфейса и обеспечить взаимодействие с пользователем. Главным образом <input> предназначен для создания текстовых полей, различных кнопок, переключателей и флажков. Хотя элемент <input> не требуется помещать внутрь контейнера <form>, определяющего форму, но если введенные пользователем данные должны быть отправлены на сервер, где их обрабатывает серверная программа, то указывать <form> обязательно. То же самое обстоит и в случае обработки данных с помощью клиентских приложений, например, скриптов на языке JavaScript.

**Основной атрибут тега <input>, определяющий вид элемента — type.** Он позволяет задавать следующие элементы формы: текстовое поле (text), поле с паролем (password), переключатель (radio), флажок (checkbox), скрытое поле (hidden), кнопка (button), кнопка для отправки формы (submit), кнопка для очистки формы (reset), поле для отправки файла (file) и кнопка с изображением (image). Для каждого элемента существует свой список атрибутов, которые определяют его вид и характеристики. Кроме того, в HTML5 добавлено еще более десятка новых элементов.

<input type="text" name="inp" value=" same value " />

Имеет следующие обязательные атрибуты:

* name-имя элемента формы
* value–значение,которое будет передано на сервер
* type-тип поля

**Атрибут type**

<input type="text">-элемент формы, создает текстовое поле для ввода.

<input type="password">-элемент формы, который реализует текстовое поле для ввода пароля. При этом все символы внутри заменяются точками.

<input type="checkbox">-элемент формы, который реализует квадратное поле с галочкой. Имеет два состояния -выбранный и не выбранный.

<inputtype="radio">-элемент формы, который реализует переключатель. Имеет два состояния -выбранный и не выбранный.

**Способы создания кнопок**

Есть несколько основных способов реализовать кнопки в разметке. Первый –использовать тег <input>с особенным значением атрибута type(submit, reset):

1. type="submit"-элемент формы, который реализует кнопку отправки данных на сервер. При нажатии формируется запрос и отправляется файлу, указанному в атрибуте action="".
2. type="reset" -элемент формы, который реализует кнопку очистки полей формы. При этом поля не просто очищаются от внесенных данных,но и восстанавливают значения по умолчанию.

Второй –создание тега <input>со значением атрибута name= "button". При этом созданная кнопка может реализовать любой функционал, указанный в ее дополнительных атрибутах обработчиков событий.

<input type="button" name="button" value="Do" onclick="ToDo()"

**Тег <select>…</select>**

Тег <select>-контейнер, содержащий в себе описание выпадающего списка. Каждый элемент списка описан внутри с помощью отдельного тега <option>.

Содержимое тега <option>будет отображено пользователям в качестве элементов выпадающего списка, но на сервер будут отправлены именно значения атрибутов value и name.

**<select name="select">**

**<option value="f1">Текст1</option>**

**<option value="f2">Текст2</option>**

**<option value="f3">Текст3</option>**

**<optionvalue="f4">Текст4</option>**

**<optionvalue="f5">Текст5</option>**

**</select>**

Если Вам необходимо реализовать список с многозначным выбором, то следует указать атрибут

multiple=“multiple”

**Параметры"hidden"и"file"**

Существует несколько особенных типов у тега <input>:

type = "hidden" -элемент формы, который не отображается пользователю, при этом передается на сервер при отправке. Используется для передачи параметров,которые не должны быть видны либо изменяться пользователем.

type = "file"-элемент формы, предназначенный для создания поля выбора загрузки файла. Реализован в каждом браузере по-разному.

**Тег <fieldset>**

Тег <fieldset>предназначен для группирования элементов формы. Такая группировка облегчает работу с формами, содержащими большое число данных.

Тег<legend> применяется для создания заголовка группы элементов формы, которая определяется с помощью тега <fieldset>.

<fieldset>

<legend>Группировка элементов </legend>

…

Любое содержимое формы

…

</fieldset>

**Тег <textarea>…</textarea>**

Тег<textarea> предназначенный для создания многострочного поля для ввода. Имеет специальные атрибуты rows и cols, которые соответственно указывают его высоту и ширину.

<textarea cols="100" rows="4">content</textarea>

**Атрибут cheсked**

Отвечает за состояние элемента, выбран он или нет.

Используется для двух типов тега <input>, при этом необходимо помнить об особенностях их группировки:

**Checkbox**–при группировке элементов с помощью ***одинакового значения атрибута name***, в активное состояние могут быть установлены одновременно несколько элементов группы.

**Radio**–при группировке элементов с помощью ***одинакового значения атрибута name***, в активное состояние может быть установлен одновременно только один элемент группы.

**Атрибут selected**

Атрибут ***selected***–атрибут, который устанавливает определенный элемент выпадающего списка,как значение по умолчанию.

Атрибут ***disabled***-указывает что элемент формы будет отображаться пользователю не активным и пользователь его не сможет изменить.

Атрибут ***readonly*** позволяет сделать элемент не восприимчивым к изменениям пользователя.

Для управления размерами текстовых полей и установки максимально допустимых значений в разметке, предусмотрены специальные атрибуты:

Атрибут ***size*** используется для указания размеров текстового поля для ввода.

Атрибут ***maxlength*** используется для установки максимально допустимого значение для текстового поля.

**Задание в классе**

**Создайте форму со следующими полями:**

*Текстовые поля по 50 символов с максимальной длиной 50*

Имя

Фамилия

Отчество

*Радио-переключатель*

Пол(М/Ж)

*Выпадающий список*

Выбор курса(C# asp.net core, Java, Angular )

*Группа флажков*

Выбор пройденных курсов (C++, C# базовый, SQL Server, WinForm, WPF, HTML)

*Многострочное поле для ввода*

Дополнительные сведения

*Кнопки*

Кнопки отправки на сервер и очистки полей