

# HTML & CSS



Forms

# HTML & CSS

## Автор курса



Александр Петрик  
MS



MCID: E754-7235

# HTML & CSS

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство Вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

# HTML & CSS

Tema

Forms

# HyperText Markup Language

## Тег `<form>...</form>`

Тег `<form>` является контейнером для формы, размещенной 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.

У него есть следующие атрибуты:

1. **action** – ссылка на файл, который должен будет обработать данные формы.
2. **method** – метод запроса на сервер, есть два основных значения: GET и POST. По умолчанию GET.
3. **target** – указывает, как будут отображены результаты, в текущем окне либо новой вкладке.
4. **autocomplete** – контролирует автозаполнение форм, имеет проблемы с кроссбраузерностью.

# HyperText Markup Language

## GET и POST

### GET

Этот метод является одним из самых распространенных и предназначен для получения требуемой информации и передачи данных в адресной строке.

Пары «имя=значение» присоединяются в этом случае к адресу после вопросительного знака и разделяются между собой амперсандом (символ &).

### POST

Метод POST посылает на сервер данные в запросе браузера. Это позволяет отправлять большее количество данных, чем доступно методу GET, поскольку у него установлено ограничение в 4 Кб.

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

# HyperText Markup Language

## Элементы формы, 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.

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

Имеет следующие обязательные атрибуты:

1. **name** – имя элемента формы;
2. **value** – значение, которое будет передано на сервер;
3. **type** – тип поля.

# HyperText Markup Language

## Типы элементов формы, атрибут type

1. **input type="text"** – элемент формы, создает текстовое поле для ввода.

2. **input type="password"** – элемент формы, который реализует текстовое поле для ввода пароля. При этом все символы внутри заменяются на маркер "точка".

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

I like Bike: ☒

I like Airplane: ☐

4. **input type="radio"** – элемент формы, который реализует radio-переключатель. Имеет два состояния – выбранный и не выбранный.

I will travel to London on Bike: ☒

I will travel to London on Airplane: ☐



# HyperText Markup Language

## Создание кнопок на форме

Есть несколько основных способов реализовать кнопки в разметке. Первый – использовать элементы `input` с особым 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 элемента `input` со значением атрибута `type = 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()" />
```

# HyperText Markup Language

## Выпадающий список, 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>  
  <option value="f4">Текст4</option>  
  <option value="f5">Текст5</option>  
</select>
```

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

**multiple="multiple"**

# HyperText Markup Language

## Различные типы тега <input>

Также есть еще несколько особенных типов элементов `input`:

1. `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.
2. `type="file"` – элемент формы, предназначенный для создания поля выбора загрузки файла. Реализован в каждом браузере по-разному.

Еще ряд очень эффектных и полезных полей для ввода вышел в спецификации HTML5, но в связи с тем, что реализация еще не закончена во многих браузерах, следует повременить с их использованием.

# HyperText Markup Language

## Группировка элементов формы, 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>**

# HyperText Markup Language

Тег `<textarea>...</textarea>`

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

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

При этом, его реализация в разных браузерах тоже отличается. В некоторых, он приобретает возможность изменять размер, расширяться, в некоторых – отличается визуальным видом.

# HyperText Markup Language

**Attributes**

# HyperText Markup Language

## Устаревшие атрибуты элементов формы

В связи с развитием HTML некоторые элементы, а также атрибуты устарели и потеряли свою значимость, но так как сайты не «совершенствуются» самостоятельно, они еще воспринимаются в некоторых браузерах.

Ниже предоставлен список, некоторых из них:

1. **Align** – атрибут, отвечающий за выравнивание, работает не во всех полях для ввода.
2. Атрибуты **alt** и **title** отвечают за значение, которое пользователю браузер будет выдавать как подсказку при наведении на элемент, с текстом, написанном в этих атрибутах.
3. Атрибут **border** – устанавливает рамку вокруг элементов, в текущих версиях браузеров не работает.

# HyperText Markup Language

## Атрибут checked

Атрибут **checked** – отвечает за состояние элемента, выбран он или нет.

Используется для двух типов элемента `input`, при этом необходимо помнить об особенностях их группировки:

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

I like Bike: ☒  
I like Airplane: ☒  
I like Car: ☐

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

I will travel to London on Bike: ☒  
I will travel to London on Airplane: ☐  
I will travel to London on Car: ☐



# HyperText Markup Language

## Атрибут selected

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

```
<select name="select">
```

```
  <option value="film001">2 smoking guns</option>
```

```
  <option value="film002">Bad Boys</option>
```

```
  <option value="film003">Revenge</option>
```

```
  <option value="film004" selected="selected">Crow</option>
```

```
  <option value="film005">Dead Man Walking</option>
```

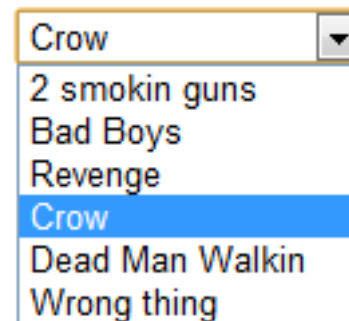
```
  <option value="film006">Wrong thing</option>
```

```
</select>
```

При загрузке страницы

A screenshot of a web browser showing a dropdown menu. The menu is closed, and the text 'Crow' is visible in the input field. The dropdown arrow is on the right side.

При открытии списка

A screenshot of a web browser showing a dropdown menu with the list open. The list contains the following items: 'Crow', '2 smokin guns', 'Bad Boys', 'Revenge', 'Dead Man Walkin', and 'Wrong thing'. The 'Crow' option is highlighted with a blue background, indicating it is the selected option.

# HyperText Markup Language

## Атрибуты disabled и readonly

Атрибут **disabled** – указывает на то, что элемент формы должен отображаться пользователю не активным и быть не восприимчивым к изменениям пользователя.

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

Отличие между ними зависит от браузера, а точнее – от реализации элементов с данными атрибутами.

name(readonly):

name(disabled):

# HyperText Markup Language

## Атрибуты size и maxlength

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

1. Атрибут **size** используется для указания размеров текстового поля для ввода.
2. Атрибут **maxlength** используется для того, чтобы задать максимально допустимое значение для текстового поля.

```
<input type="text" name="first_name" value="Mickey" size="120" maxlength="100" />
```



← 120 – длина поля,  
а максимальная длина вводимого значения – 100 →

# Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.

Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics



# Проверка знаний

TestProvider.com

TestProvider

Мы помогаем людям оценить себя

Главная Услуги и цены Центр Тестирования Поддержка О нас

Регистрация Войти

Поиск сертификата

Мы в социальных сетях

## Тестирование

Языки программирования и информационные технологии

**Microsoft**

C# ASP.NET MVC JavaScript Patterns OF Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Добро пожаловать на TestProvider.com!

Сайт перенесен на новую облачную платформу с использованием системы единой авторизации Single Sign On. Если вы хотите восстановить статистику по предыдущим экзаменам обратитесь в [службу поддержки](#). Для восстановления информации с предыдущей версии сайта, просба написать в службу поддержки Ваш старый и новый логины.

ITVDN PROMETRIC TEST CENTER CyberBionic Microsoft Partner Windows Azure Cloud Partner EBA

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](http://TestProvider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



## Q&A

# Информационный видеосервис для разработчиков программного обеспечения

