

# Веб-формы — Работа с пользовательскими данными

Этот модуль содержит серию статей, которые помогут вам освоить основы веб-форм. Веб-формы — очень мощный инструмент для взаимодействия с пользователями — чаще всего они используются для сбора данных от пользователей или предоставления им возможности управлять пользовательским интерфейсом. Однако по историческим и техническим причинам не всегда очевидно, как использовать их в полной мере. В перечисленных ниже статьях мы рассмотрим все основные аспекты веб-форм, включая разметку их HTML-структуры, стилизацию элементов управления формы, проверку данных формы и отправку данных на сервер.

Хотите стать фронтенд-разработчиком? Мы составили курс, который включает в себя всю необходимую информацию, необходимую для достижения вашей цели.

Начать

**Предварительные условия** Прежде чем приступить к этому модулю, вы должны хотя бы изучить наше введение в HTML . На этом этапе вы должны найти вводные руководства простыми для понимания, а также иметь возможность использовать наше базовое руководство по элементам управления формами.

Однако для освоения форм требуется нечто большее, чем просто знание HTML — вам также необходимо изучить некоторые специальные приемы стилизации элементов управления формы, а также некоторые знания сценариев необходимы для обработки таких вещей, как проверка и создание настраиваемых элементов управления формы. Поэтому, прежде чем вы посмотрите на другие разделы, перечисленные ниже, мы рекомендуем вам сначала уйти и немного изучить CSS и JavaScript .

Вышеприведенный текст является хорошим показателем того, почему мы поместили веб-формы в отдельный автономный модуль, а не пытаемся смешивать его части с темами HTML, CSS и JavaScript — элементы формы более сложны, чем большинство других HTML. элементы, и они также требуют тесного сочетания связанных методов CSS и JavaScript, чтобы получить от них максимальную отдачу.

**Примечание.** Если вы работаете на компьютере/планшете/другом устройстве, на котором у вас нет возможности создавать собственные файлы, вы можете попробовать (большинство) примеров кода в онлайн-программах кодирования, таких как JSBin или Glitch .

**Вводные руководства** Ваша первая форма В первой статье из нашей серии рассказывается о вашем первом опыте создания веб-формы, в том числе о разработке простой формы, ее реализации с использованием правильных элементов HTML, добавлении очень простых стилей с помощью CSS и о том, как данные отправляются на сервер.

**Как структурировать веб-форму** Избавившись от основ, теперь мы более подробно рассмотрим элементы, используемые для придания структуры и значения различным частям формы.

**Различные элементы управления формой** Основные собственные элементы управления формой `<input>` Мы начинаем этот раздел с подробного рассмотрения функциональности исходных типов HTML и того, какие параметры доступны для сбора различных типов данных.

**Типы ввода HTML5** Здесь мы продолжаем наше глубокое погружение в `<input>` элемент, рассматривая дополнительные типы ввода, предоставленные при выпуске HTML5, а также различные элементы управления пользовательского интерфейса и улучшения сбора данных, которые они предоставляют. Дополнительно смотрим на `<output>` элемент.

**Другие элементы управления формой** Далее мы рассмотрим все элементы `<input>` управления, не относящиеся к форме, и связанные с ними инструменты, такие как `<select>`, `<textarea>`, `<meter>` и `<progress>`.

**Руководства по стилю форм** Стилизация веб-форм В этой статье представлено введение в стилизацию форм с помощью CSS, включая все основы, которые вам могут понадобиться для базовых задач по стилизации.

**Расширенный стиль формы** Здесь мы рассмотрим некоторые более продвинутые методы стилизации форм, которые необходимо использовать при работе с некоторыми элементами формы, которые сложнее стилизовать.

**Псевдоклассы пользовательского интерфейса** Введение в псевдоклассы пользовательского интерфейса, позволяющие нацеливать элементы управления формы HTML на основе их текущего состояния.

**Проверка и отправка данных формы** Проверка формы на стороне клиента Отправки данных недостаточно — нам также необходимо убедиться, что данные, которые пользователи вводят в формы, имеют правильный формат для их успешной обработки и не нарушают работу наших приложений. Мы также хотим помочь нашим пользователям правильно заполнять наши формы и не расстраиваться при попытке использовать наши приложения. Проверка формы помогает нам достичь этих целей — в этой статье рассказывается, что вам нужно знать.

**Отправка данных формы** В этой статье рассматривается, что происходит, когда пользователь отправляет форму — куда попадают данные и как мы обрабатываем их, когда они туда попадают? Мы также рассмотрим некоторые проблемы безопасности, связанные с отправкой данных формы.

**Расширенные статьи** Следующие статьи не являются обязательными для пути обучения, но они окажутся интересными и полезными, если вы освоите описанные выше методы и захотите узнать больше.

**Как создавать собственные элементы управления формы** Вы столкнетесь с некоторыми случаями, когда собственные виджеты форм просто не предоставляют то, что вам нужно, например, из-за стиля или функциональности. В таких случаях вам может понадобиться создать собственный виджет формы из необработанного HTML. В этой статье объясняется, как это сделать, и соображения, которые необходимо учитывать при этом, с практическим примером.

**Отправка форм через JavaScript** В этой статье рассматриваются способы использования формы для сборки HTTP-запроса и отправки его с помощью пользовательского JavaScript, а не стандартной отправки формы. Также рассматривается, почему вы хотели бы это сделать, и последствия этого. (См. также [Использование объектов FormData](#) .)

**Таблица совместимости свойств CSS для элементов управления формы** Эта последняя статья содержит удобную справочную информацию, позволяющую узнать, какие свойства CSS

совместимы с элементами формы.

From:

<https://vladpolskiy.ru/> - **book51.ru**

Permanent link:

<https://vladpolskiy.ru/doku.php?id=software:development:web:docs:learn:forms>

Last update: **2023/08/22 21:33**

