#### Формы, элементы управления

Навигация по уроку

Событие: submit

Метод: submit

Задачи (1)

Комментарии

Поделиться



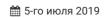




Редактировать на GitHub



→ Браузер: документ, события, интерфейсы → Формы, элементы управления



# Отправка формы: событие и метод submit

При отправке формы срабатывает событие submit, оно обычно используется для проверки (валидации) формы перед её отправкой на сервер или для предотвращения отправки и обработки её с помощью JavaScript.

Metog form.submit() позволяет инициировать отправку формы из JavaScript. Мы можем использовать его для динамического создания и отправки наших собственных форм на сервер.

Давайте посмотрим на них подробнее.

### Событие: submit

Есть два основных способа отправить форму:

- 1. Первый нажать кнопку <input type="submit"> или <input type="image">.
- 2. Второй нажать Enter, находясь на каком-нибудь поле.

Оба действия сгенерируют событие submit на форме. Обработчик может проверить данные, и если есть ошибки, показать их и вызвать event.preventDefault(), тогда форма не будет отправлена на сервер.

В примере ниже:



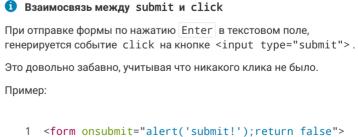
- 1. Перейдите в текстовое поле и нажмите Enter.
- 2. Нажмите <input type="submit">.

Оба действия показывают alert и форма не отправится благодаря return false:

```
<form onsubmit="alert('submit!');return false">
```

- Первый пример: нажмите Enter: <input type="text" value
- Второй пример: нажмите на кнопку "Отправить": <input 3
- </form>

Первый пример: нажмите Enter: Текст Второй пример: нажмите на кнопку "Отправить": Отправить



```
<input type="text" size="30" value="Установите фс
```

<input type="submit" value="Отправить" onclick="a

</form>

Установите фокус здесь и нажмите Enter

Раздел

#### Формы, элементы управления

Навигация по уроку

Событие: submit

Метод: submit

Задачи (1)

Комментарии

Поделиться







Редактировать на GitHub

### Метод: submit

Чтобы отправить форму на сервер вручную, мы можем вызвать метод form.submit().

При этом событие submit не генерируется. Предполагается, что если программист вызывает метод form.submit(), то он уже выполнил всю соответствующую обработку.

Иногда это используют для генерации формы и отправки её вручную, например так:

```
1 let form = document.createElement('form');
2 form.action = 'https://google.com/search';
3 form.method = 'GET';
4
5 form.innerHTML = '<input name="g" value="test">';
6
7
   // перед отправкой формы, её нужно вставить в документ
8
   document.body.append(form);
10 form.submit();
```

## Задачи

## Модальное диалоговое окно с формой 💆

важность: 5

Создайте функцию showPrompt(html, callback), которая выводит форму с сообщением ( html ), полем ввода и кнопками ОК/ОТМЕНА.

- Пользователь должен ввести что-то в текстовое поле и нажать Enter или кнопку «ОК», после чего должна вызываться функция callback(value) со значением поля.
- Если пользователь нажимает Еsc или кнопку «ОТМЕНА», тогда вызывается callback(null).

В обоих случаях нужно завершить процесс ввода и закрыть диалоговое окно с формой.

#### Требования:

- Форма должна быть в центре окна.
- Форма является модальным окном, это значит, что никакое взаимодействие с остальной частью страницы невозможно, пока пользователь не закроет его.
- При показе формы, фокус должен находиться сразу внутри <input>.
- Клавиши Tab / Shift+Tab должны переключать фокус между полями формы, не позволяя ему переходить к другим элементам страницы.

Пример использования:

```
1 showPrompt("Введите что-нибудь<br>>...умное :)", functio
2
    alert(value);
3 });
```

Демо во фрейме:

## Кликните на кнопку ниже

Кликните, чтобы увидеть форму

P.S. HTML/CSS исходного кода к этой задаче содержит форму с фиксированным позиционированием, но вы должны сделать её модальной.





 $\equiv$ 

Å



Открыть песочницу для задачи.

( решение

Раздел

Формы, элементы управления

Навигация по уроку

Событие: submit

Метод: submit

Задачи (1)

Комментарии

Поделиться



Редактировать на GitHub





Проводим курсы по JavaScript и фреймворкам.





перед тем как писать...

© 2007—2020 Илья Кантор | о проекте | связаться с нами | пользовательское соглашение | политика конфи



