

Раздел

[Формы, элементы управления](#)

Навигация по уроку

Событие: change

Событие: input

События: cut, copy, paste

Итого

Задачи (1)

Комментарии

Поделиться



Редактировать на GitHub

[🏠 → Браузер: документ, события, интерфейсы](#)
[→ Формы, элементы управления](#)

4-го июля 2019

События: change, input, cut, copy, paste

Давайте рассмотрим различные события, сопутствующие обновлению данных.

Событие: change

Событие `change` срабатывает по окончании изменения элемента.

Для текстовых `<input>` это означает, что событие происходит при потере фокуса.

Пока мы печатаем в текстовом поле в примере ниже, событие не происходит. Но когда мы перемещаем фокус в другое место, например, нажимая на кнопку, то произойдёт событие `change`:

```
1 <input type="text" onchange="alert(this.value)">
2 <input type="button" value="Button">
```

Button

Для других элементов: `select`, `input type=checkbox/radio` событие запускается сразу после изменения значения:

```
1 <select onchange="alert(this.value)">
2   <option value="">Выберите что-нибудь</option>
3   <option value="1">Вариант 1</option>
4   <option value="2">Вариант 2</option>
5   <option value="3">Вариант 3</option>
6 </select>
```

Выберите что-нибудь ▼

Событие: input

Событие `input` срабатывает каждый раз при изменении значения.

В отличие от событий клавиатуры, оно работает при любых изменениях значений, даже если они не связаны с клавиатурными действиями: вставка с помощью мыши или распознавание речи при диктовке текста.

Например:

```
1 <input type="text" id="input"> oninput: <span id="result">
2 <script>
3   input.oninput = function() {
4     result.innerHTML = input.value;
5   };
6 </script>
```

oninput:

Если мы хотим обрабатывать каждое изменение в `<input>`, то это событие является лучшим выбором.

С другой стороны, событие `input` не происходит при вводе с клавиатуры или иных действиях, если при этом не меняется значение в текстовом поле, т.е. нажатия клавиш `↵`, `⬅` и подобных при фокусе на текстовом поле не вызовут это событие.

Раздел

Формы, элементы управления

Навигация по уроку

Событие: change

Событие: input

События: cut, copy, paste

Итого

Задачи (1)

Комментарии

Поделиться



Редактировать на GitHub



❗ Нельзя ничего предотвратить в oninput

Событие input происходит после изменения значения.

Поэтому мы не можем использовать event.preventDefault() там – будет уже слишком поздно, никакого эффекта не будет.

События: cut, copy, paste

Эти события происходят при вырезании/копировании/вставке данных.

Они относятся к классу ClipboardEvent и обеспечивают доступ к копируемому/вставляемому данным.

Мы также можем использовать event.preventDefault() для предотвращения действия по умолчанию, и в итоге ничего не скопируется/не вставится.

Например, код, приведённый ниже, предотвращает все подобные события и показывает, что мы пытаемся вырезать/копировать/вставить:

```
1 <input type="text" id="input">
2 <script>
3   input.oncut = input.oncopy = input.onpaste = function
4     alert(event.type + ' - ' + event.clipboardData.getD
5     return false;
6   };
7 </script>
```

Технически, мы можем скопировать/вставить всё. Например, мы можем скопировать файл из файловой системы и вставить его.

Существует список методов в спецификации для работы с различными типами данных, чтения/записи в буфер обмена.

Но обратите внимание, что буфер обмена работает глобально, на уровне ОС. Большинство браузеров в целях безопасности разрешают доступ на чтение/запись в буфер обмена только в рамках определённых действий пользователя, к примеру, в обработчиках событий onclick.

Также запрещается генерировать «пользовательские» события буфера обмена при помощи dispatchEvent во всех браузерах, кроме Firefox.

Итого

События изменения данных:

Событие	Описание	Особенности
change	Значение было изменено.	Для текстовых полей срабатывает при потере фокуса.
input	Срабатывает при каждом изменении значения.	Запускается немедленно, в отличие от change.
cut/copy/paste	Действия по вырезанию/копированию/вставке.	Действие можно предотвратить. Свойство event.clipboardData предоставляет доступ на чтение/запись в буфер обмена...

✅ Задачи

Депозитный калькулятор [↗](#)

Создайте интерфейс, позволяющий ввести сумму банковского вклада и процент, а затем рассчитать, какая это будет сумма через заданный промежуток времени.



Демо-версия:

Депозитный калькулятор.

Первоначальный депозит

10000

Срок вклада?

12 (год)

Годовая процентная ставка?

5

Было: Станет:

10000

10500

Любое изменение введённых данных должно быть обработано немедленно.

Формула:

```
1 // initial: начальная сумма денег
2 // interest: проценты, например, 0.05 означает 5% в год
3 // years: сколько лет ждать
4 let result = Math.round(initial * (1 + interest * years
```

[Открыть песочницу для задачи.](#)



решение



Проводим [курсы по JavaScript и фреймворкам.](#)



Комментарии

перед тем как писать...

Раздел

[Формы, элементы управления](#)

Навигация по уроку

Событие: change

Событие: input

События: cut, copy, paste

Итого

Задачи (1)

Комментарии

Поделиться



[Редактировать на GitHub](#)