

Основы JavaScript

Объект события. всплытие. Делегирование

Объект события

— — —

При обработке события могут понадобиться детали (координаты курсора, код нажатой кнопки).

Детали произошедшего браузер записывает в так называемый «объект события» (Event Object), который передается первым аргументом в функцию-обработчик.

Справка по Event Object (ENG):

https://www.w3schools.com/jsref/obj_events.asp

Объект события

Пример:

```
var block = document.querySelector("#hello");
```

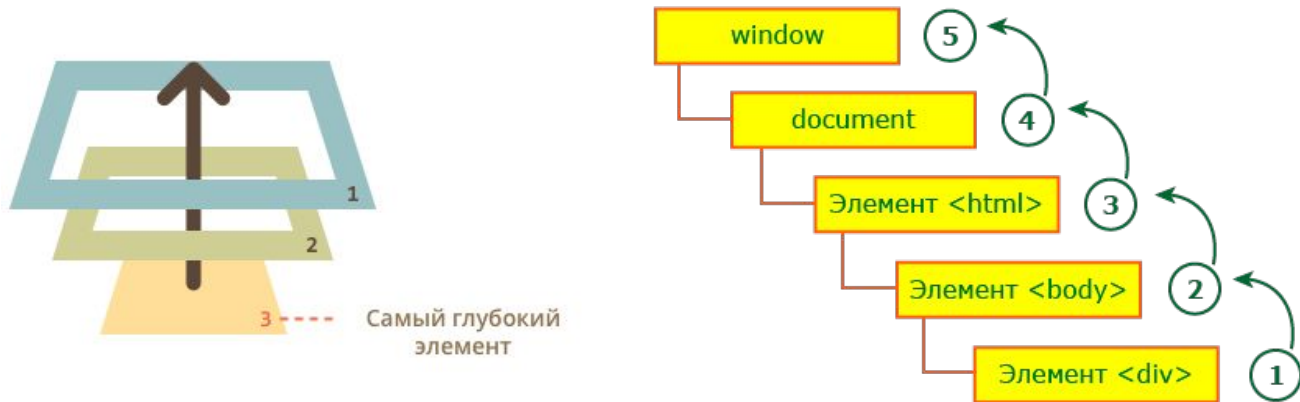
```
block.addEventListener("click", function(event) {  
    var event = event || window.event;  
  
    console.log(event.type);  
    console.log(event.currentTarget);  
    console.log(event.clientX + " , " + event.clientY);  
});
```

Для наших любимых
IE8-

Всплытие

— — —

При наступлении события обработчики сначала срабатывают на самом вложенном элементе, затем на его родителе, затем выше и так далее, вверх по цепочке вложенности.



Всплытие (Bubbling)

— — —

Пример:

```
<form onclick="alert('form')">FORM  
  <div onclick="alert('div')">DIV  
    <p onclick="alert('p')">P</p>  
  </div>  
</form>
```

Клик по абзацу приведет к вызову не только обработчика абзаца, а и к вызову обработчиков его родителей по цепочке.

Demo

Всплытие



Целевой элемент

— — —

Самый глубокий элемент, который вызывает событие, называется «целевым» или «исходным» элементом и доступен как `event.target`.

Отличия `this` (`event.currentTarget`) и `event.target`:

- `event.target` – это **исходный элемент**, на котором произошло событие, в процессе всплытия он неизменен.
- `this` – это **текущий элемент**, до которого дошло всплытие, на нём сейчас выполняется обработчик.

Прекращение всплытия

— — —

Всплытие идёт прямо вверх. Обычно событие будет всплывать вверх и вверх, до элемента `<html>`, а затем до `document`, а иногда даже до `window`, вызывая все обработчики на своем пути.

Но любой промежуточный обработчик может решить, что событие полностью обработано, и остановить всплытие.

Для остановки всплытия нужно вызвать метод `event.stopPropagation()`.

Делегирование

— — —

Всплытие событий позволяет реализовать один из самых важных приемов разработки – *делегирование*.

Он заключается в том, что если у нас есть много элементов, события на которых нужно обрабатывать похожим образом, то вместо того, чтобы назначать обработчик каждому – мы ставим один обработчик на их общего предка. Из него можно получить целевой элемент `event.target`, понять на каком именно потомке произошло событие и обработать его.

Делегирование

— — —

Делегирование это один из самых полезных приемов для работы с DOM. Это прием отлично подходит, при наличии множества элементов, обработка которых очень схожа.

Алгоритм:

1. Добавляем обработчик на контейнер
2. В обработчике получаем `event.target`
3. Если `event.target` или один из его родителей в контейнере (`this`) – интересующий нас элемент – обработать его

Demo

Делегирование



Материал для самостоятельного рассмотрения

— — —

- [Погружение \(Перехват\)](#)
- [Генерация событий](#)

