



Занятие №19

Практика по обработке событий



Классы событий

События можно поделить на несколько групп:

- события мыши
- события клавиатуры
- события DOM
- события элементов управления

Объект события

В функцию - обработчик события в момент срабатывания передаётся объект события.

Объект события - это объект с информацией о произошедшем событии.

В зависимости от класса события объект несет разную информацию.

Работа с объектом события

```
elem.onclick = function(ev){  
  console.log(ev);  
}
```

`ev` - имя параметра указанного для функции, можно заменить любым другим именем. Этот параметр и будет объектом события. Если вы задаете событие с помощью атрибутов в HTML, то объект события будет доступен с помощью переменной `event`.

```
<button onclick="console.log(event)">Press</button>
```



Общие свойства и методы объекта события

Отмена действия по умолчанию

За отмену действия по умолчанию отвечает метод `preventDefault` объекта события.

```
elem.onclick = function(ev){  
    ev.preventDefault();  
}
```

Отмена всплытия

Всплытие события - это ситуация, когда событие одного элемента, происходит и с другим. Рассмотрим пример:

```
<div id="div" onclick="alert('div')">  
  <p onclick="alert('p')">  
    <button onclick="alert('button')">Press</button>  
  </p>  
</div>
```

Так как кнопка находится внутри тега `p`, который находится внутри `div`, то по щелчку по кнопке событие `click` сработает и у тега `p`, и у тега `div`. Данное событие сначала происходит у самого “глубокого” элемента, а затем поднимается выше (поэтому всплытие). Метод `stopPropagation` у объекта события отменяет всплытие, то есть дальше событие не всплывает.

Пример

```
<div id="div" onclick="alert('div')">
```

```
  <p onclick="alert('p')">
```

```
    <button onclick="event.stopPropagation();  
alert('button')">Press</button>
```

```
  </p>
```

```
</div>
```


Свойства target и currentTarget

Оба свойства ссылаются на источник события, однако `target` - это элемент, на котором событие возникло, а `currentTarget` - элемент, на который добавили обработчик. `currentTarget` появляется, когда есть всплытие события.

Далее указан код, на котором можно отследить разницу.

```
<p id="p">  
  <button id="btn">Press</button>  
</p>
```

Другие свойства объекта события

- `timeStamp` - время в мс, прошедших с момента загрузки страницы до события
- `type` - имя события



События мыши

События мыши

Дополнительные события мыши

- `dblclick` - двойной клик
- `mousedown` - кнопка мыши опущена вниз
- `mouseup` - кнопка мыши поднята вверх

Свойства объекта события мыши

Координаты:

- `pageX, pageY` - координаты от начала документа (как и `position: absolute`)
- `clientX, clientY` - координаты от окна (прокрутка не учитывается)
- `offsetX, offsetY` - координаты от начала самого элемента

Свойства объекта события мыши

Дополнительные клавиши:

- `ctrlKey` - true, если была нажата клавиша ctrl
- `altKey` - true, если была нажата клавиша alt
- `shiftKey` - true, если была нажата клавиша shift

`which` - какая кнопка была нажата (работает с `mousedown` и `mouseup`).

1 - левая, 3 - правая, 2 - колесико (средняя)



События клавиатуры

События клавиатуры

- `keydown` - клавиша клавиатуры опущена вниз, нажата (предпочтительный вариант)
- `keyup` - клавиша клавиатуры поднята вверх, вернулась в исходное состояние
- `keypress` - нажата символьная клавиша

Добавление обработчика события

Событие “нажатие клавиши” может происходить только в элементах, в которых нажатие клавиши работает (элементы управления), либо его можно добавить на глобальный объект [window](#)

Основные свойства объекта события

- `key` - обозначение нажатой клавиши в виде строки (например, "w")
- `code` - код нажатой клавиши в виде строки (например, "KeyW")
- `keyCode` - численный код нажатой клавиши
- `altKey`, `ctrlKey`, `shiftKey` - по аналогии с событиями мыши

События загрузки страницы

`window.onload = fn;`

- Срабатывает в момент когда ВСЕ данные загружены (включая картинки, файлы css)
- Можно добавлять обработчик в свойства и через `addEventListener`

Есть у объекта `window`

`document.addEventListener("DOMContentLoaded", fn);`

- Срабатывает в момент, когда загружено дерево DOM, без картинок и стилей
- Добавляется только через `addEventListener`
- Есть у объекта `document`

Задания

1. Напиши калькулятор с функционалом для кнопок, а также для ввода с клавиатуры функции.
2. Напиши функцию, которая включает (ctrl + e) и отключает (ctrl + s) темный режим на странице.
3. Напишите функцию, которая перетаскивает элемент (квадрат) на странице по клику мыши на этот элемент
4. Создать игру управление кубиком (персонажем) по нажатию клавиш



Давайте подведем итоги!
Чему мы научились?
Что мы использовали?