JS: DOM

AIT
Hi-tech team

НАШИ ПРАВИЛА

Включенная камера

Вопросы по поднятой руке

Не перебиваем друг друга

Все вопросы, не связанные с тематикой курса (орг-вопросы и т. д.), должны быть направлены куратору

Подготовьте свое рабочее окружение для возможной демонстрации экрана (закройте лишние соцсети и прочие приложения)

ПОИГРАЕМ ;)

Сколько способов создать функцию вы узнали?

Какие функции называются анонимными?

Что такое DOM?

Какие методы поиска элементов в DOM вы знаете?

ЦЕЛЬ

Узнать методы работы с DOM.

ПЛАН ЗАНЯТИЯ

- Методы работы с DOM
- Получение данных из формы

Добавление элементов

1 шаг - Создать элемент

Метод: `document.createElement("tag")`

Создание: `let el = document.createElement("div")`

2 шаг - Заполнить элемент

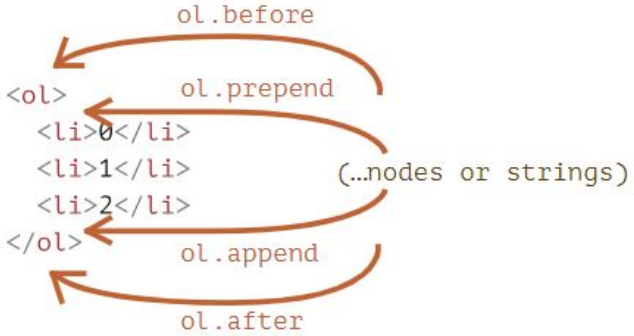
Метод: `node.textContent = “text”`

Заполнение: el.textContent = “Text”

3 шаг - Добавляем элемент в document

Методы:

- **node.append(el) – добавляет узлы или строки в конец node,**
- **node.prepend(el) – вставляет узлы или строки в начало node,**
- **node.before(el) – вставляет узлы или строки до node,**
- **node.after(el) – вставляет узлы или строки после node,**



Изменение элементов

- **textContent**

Позволяет задавать или получать текстовое содержимое элемента и его потомков.

```
let text = element.textContent

element.textContent = “Just text”
```

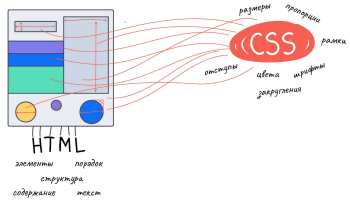
- **innerHTML**

Свойство innerHTML позволяет считать содержимое элемента в виде HTML-строки или установить новый HTML.

```html
<form>
  <label>Логин</label>
  <input type="text" id="login" />
  <div class="error">Введите логин</div>
</form>
```

```
1 const form = document.querySelector('form')
2
3 console.log(form.innerHTML)
4 // '<label>Логин</label><input type="text" id="login" /><div class="error">Вве
5
6 // Меняем содержимое новым html
7 form.innerHTML = '<div class="success">Вход выполнен</div>'
```

- **изменение стилей**



HTML DOM позволяет JavaScript изменять стиль HTML элементов.

Синтаксис:

```
document.getElementById(id).style.свойство = новый стиль
```

Применение:

```
document.getElementById("p2").style.color = "blue";
```

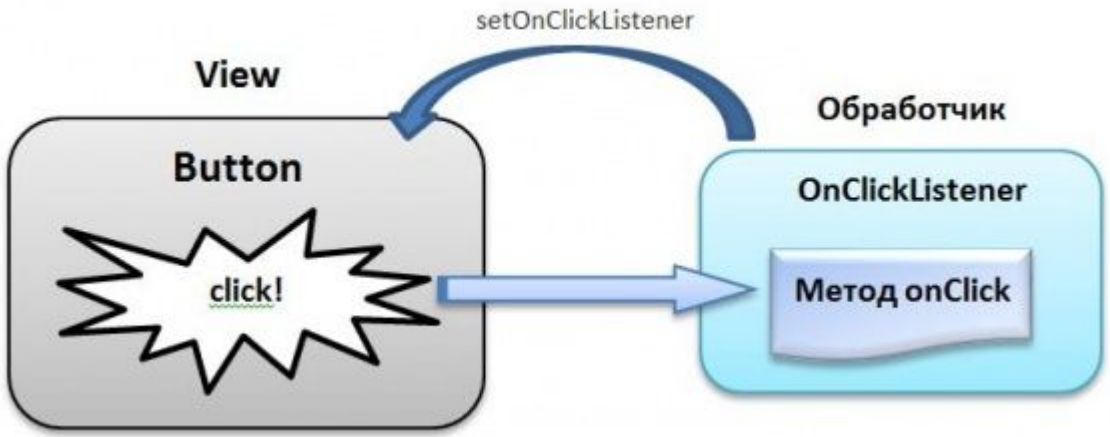
- **удаление элементов**

Для удаления узла есть метод node.remove()

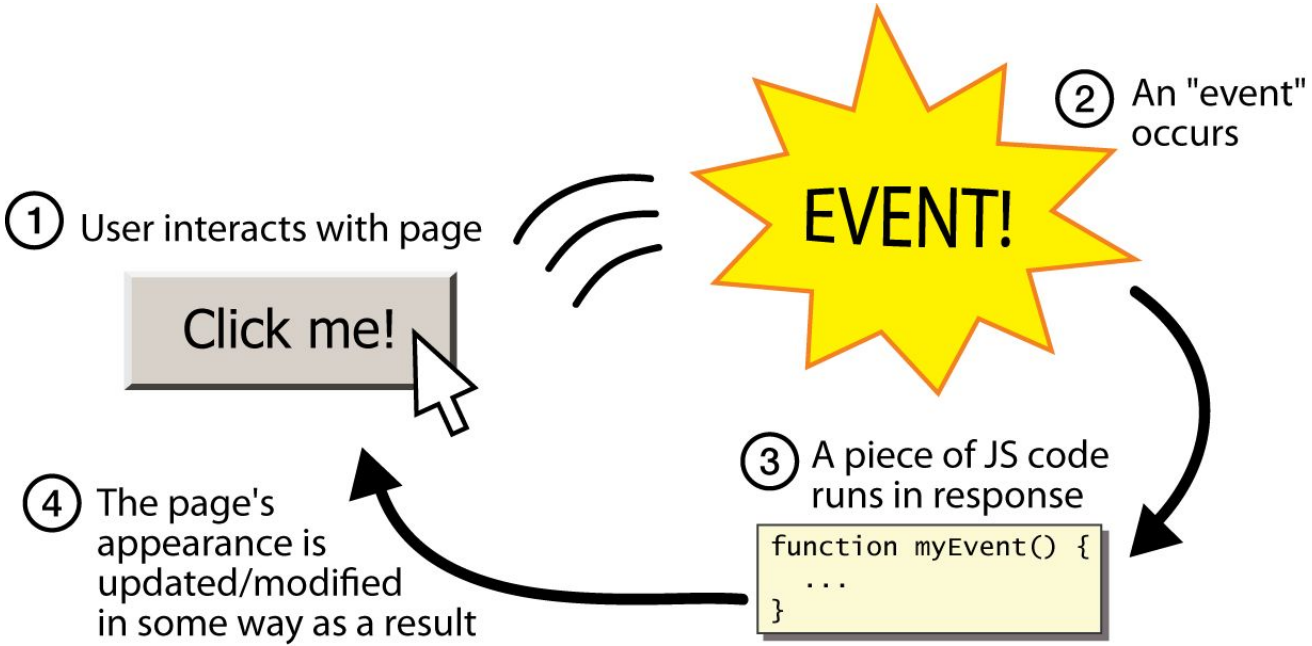
Пример использования: удаление элемента с id=”register”

```
document.getElementById("register").remove();
```

Events, event listeners

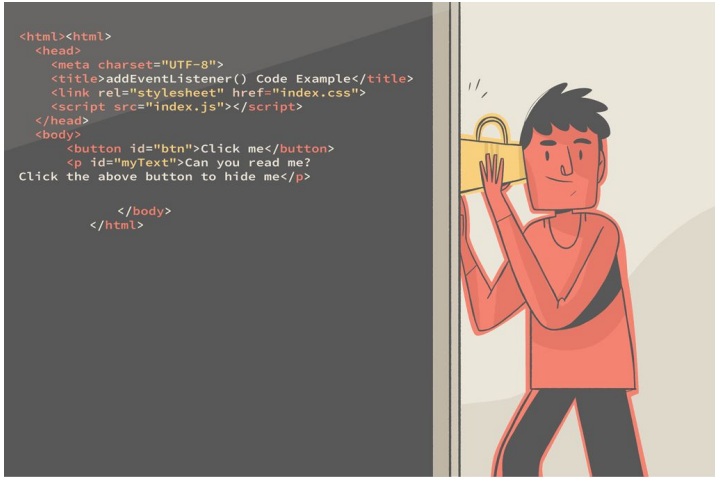


Любой DOM элемент запускает событие, когда мы с ним как-то взаимодействуем (кликаем, наводим мышь и др.). Обработчики событий в JS используются для того, чтобы реагировать на эти события.



Чтобы "повесить" обработчик событий на элемент, нужно использовать специальный метод - **addEventListener**. Этот метод принимает 2 аргумента:

1. **Тип события** (например "click").
2. Так называемую колбэк (**callback**) функцию, которая запускается после срабатывания нужного события.




```
element.addEventListener('click', handleClickFunction)
```

Пример

Найдём кнопку на странице и будем выводить сообщение в консоль, когда произошёл клик по этой кнопке.

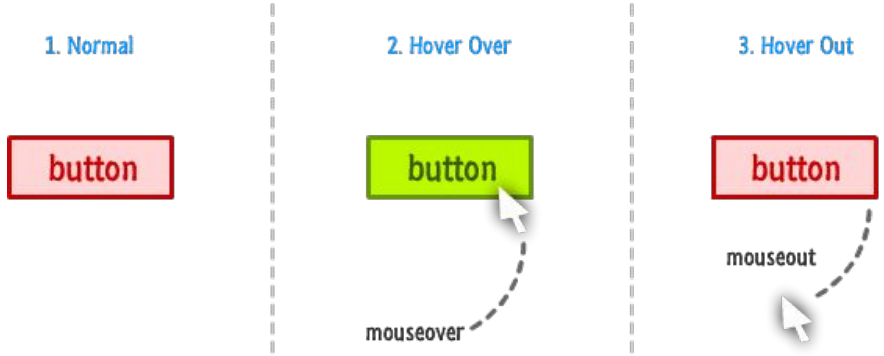
```
1 const element = document.querySelector('button')
2
3 element.addEventListener('click', function (event) {
4   console.log('Произошло событие', event.type)
5 })
```

Типы событий

eventType (первый аргумент addEventListener) - строка, содержащая название события.

Наиболее популярные события:

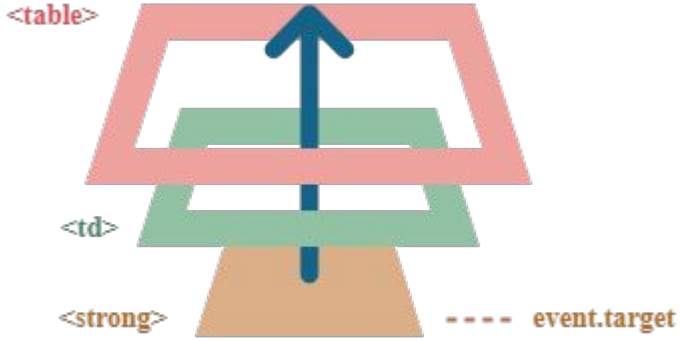
- 'click'
- 'change'
- 'submit'
- 'keydown'
- 'keyup'
- 'mousemove'
- 'mouseenter'
- 'mouseleave'



Объект Event

Когда происходит событие, браузер создаёт объект события - event, записывает в него детали и передаёт его в качестве аргумента функции-обработчику (второму аргументу addEventListener).

Объект Event - основные свойства



- **defaultPrevented** — отменено ли поведение события по умолчанию.
- **target** — ссылка на объект, которым было инициировано событие. Например, если событие произошло на поле ввода, мы получим ссылку на этот DOM элемент.
- **type** — тип события.

Объект Event - метод preventDefault

`event.preventDefault()` — предотвращает дефолтное поведение события.

Например, при нажатии на ссылку, отменить переход по адресу ссылки

Работа с формой

```
<form id="myForm">
    <label for="username">Username:</label>
    <input type="text" id="username"
name="username">

    <label for="password">Password:</label>
    <input type="password" id="password"
name="password">

    <button type="button"
onclick="getData()">Submit</button>
 </form>
```

```
function getData() {
 const form = document.getElementById('myForm');

 // Получение данных по имени
 // два варианта
 let username = form.elements["username"].value;
 let password = form.password.value;
  // Действия с данными
 console.log('Username:', username);
 console.log('Password:', password);
}
```

Ваша новая IT-профессия – Ваш новый уровень жизни

Программирование с нуля в
немецкой школе AIT TR GmbH

AIT
Hi-tech team