Цель - ознакомить вас со способами работы с DOM. Это нужно - если вы пишете автоматические интеграционные тесты. Или пытаетесь превратить данные сайта в структуру данных.

**Селекторы**

document.getElementById(id) //- получить элемент по айди - самый быстрый

document.getElementsByClassName(className)

document.getElementsByTagName(tagName)

document.querySelector(query)

**например**

document.querySelectorAll('#my-id') - получить все элементы у которых id = my-id  
document.querySelectorAll('.my-class') - получить все элементы у которых class = my-class  
document.querySelectorAll('div') - получить все элементы у которых tagName = div

document.querySelector('#my-id') - получить 1 элемент у которых id = my-id  
document.querySelector('.my-class') - получить 1 элемент у которых class = my-class  
document.querySelector('div') - получить 1 элемент у которых tagName = div

**Получить текстовое значение ноды**

node.textContent

**Получить значение аттрибута классов**

node.getAttribute(attributeName)

node.getAttribute('data-attribute')

node.getAttribute('src')

**Получить значение аттрибута классов**

element.className

**Преобразовать структуру Дом элементов в массив**

Array.from(nodes)

Array.from(document.getElementsByClassName('test')).map(it => ...)

**Изменить текстовое значение ноды**

element.textContent = 'Новое значение'

**Удалить потомок**

element.removeChild(childElement)

**Получить доступ к родительской ноде**

element.parentNode

**Создание элемента**

document.createElement(tagName)

**Изменение текста элемента**

element.innerText = 'text'

**Работа с аттрибутами**

element.hasAttribute(name) – проверяет наличие атрибута  
element.getAttribute(name) – получает значение атрибута  
element.setAttribute(name, value) – устанавливает атрибут  
element.removeAttribute(name) – удаляет атрибут

**Добавление/удаление события события**

const eventFunction = (event) => {  
console.log('clicked')  
}  
element.addEventListener('onClick', eventFunction)  
element.removeEventListener('onClick', eventFunction)

**Добававление потомка в текущий элемент**

element.appendChild(element2)

**Удалить потомка**

element.removeChild(element.childNodes[0](https://www.skillcrucial.com/en/course/48cbb267-7fbf-4a68-8785-243fd67d5a93/1584246800393/1584554135292))

**Получить предка**

element.parentNode

**Обновить текст в текстовом поле**

element.value = 'new value'