

# JavaScript

## манипуляции с DOM

Миша Ларченко / 14 июля 2020

# DOM и JavaScript

# DOM и JavaScript

The screenshot shows a web browser window titled "Learning About Nodes" with the address bar displaying `file:///Users/sammy/Documents/nodes.html`. The page content includes a header "Shark World", a paragraph "The world's leading source on related information." followed by a "shark" button, and a section "Types of Sharks" with a list of three items: "Hammerhead", "Tiger", and "Great White". Each item is enclosed in a colored box (cyan, yellow, and orange respectively).

The browser's developer console is open, showing the following JavaScript code and its execution results:

```
> p.parentNode;
< ><body>...</body>

> p.parentNode.parentNode;
< <html>
  ><head>...</head>
  ><body>...</body>
</html>

> ul.childNodes;
< >(7) [text, li, text, li, text, li, text]

> ul.firstChild.style.background = 'yellow';
Uncaught TypeError: Cannot set property 'background' of undefined
    at <anonymous>:1:32
    VM37:1

> ul.firstChild.style.background = 'yellow';
"yellow"

> for (let element of ul.children) {
  element.style.background = 'yellow';
}
"yellow"

> for (let element of p.childNodes) {
  console.log(element);
}
"The world's leading source on "
  <strong>shark</strong>
  " related information."
undefined

> document.body.children[3].lastElementChild.style.background = 'fuchsia';
"fuchsia"

> const tiger = ul.children[1];
undefined

> tiger.nextElementSibling.style.background = 'coral';
tiger.previousElementSibling.style.background = 'aquamarine';
"aquamarine"

>
```

# DOM и JavaScript

## Section 4. Manipulating elements

- `createElement()` – create a new element.
- `appendChild()` – append a node to a list of child nodes of a specified parent node.
- `textContent` – get and set the text content of a node.
- `innerHTML` – get and set the HTML content of an element.
- `innerHTML` vs. `createElement` – explain the differences between `innerHTML` and `createElement` when it comes to creating new elements.
- `DocumentFragment` – learn how to compose DOM nodes and insert them into the active DOM tree.
- `insertBefore()` – insert a new node before an existing node as a child node of a specified parent node.
- `insertAfter()` helper function – insert a new node after an existing node as a child node of a specified parent node.
- `append()` – insert a node after the last child node of a parent node.
- `prepend()` – insert a node before the first child node of a parent node.
- `insertAdjacentHTML()` – parse a text as HTML and insert the resulting nodes into the document at a specified position.
- `replaceChild()` – replace a child element by a new element.
- `cloneNode()` – clone an element and all of its descendants.
- `removeChild()` – remove child elements of a node.



# DOM и JavaScript

getElementById / getElementsByName / getElementsByTagName / getElementsByClassName

querySelector / querySelectorAll

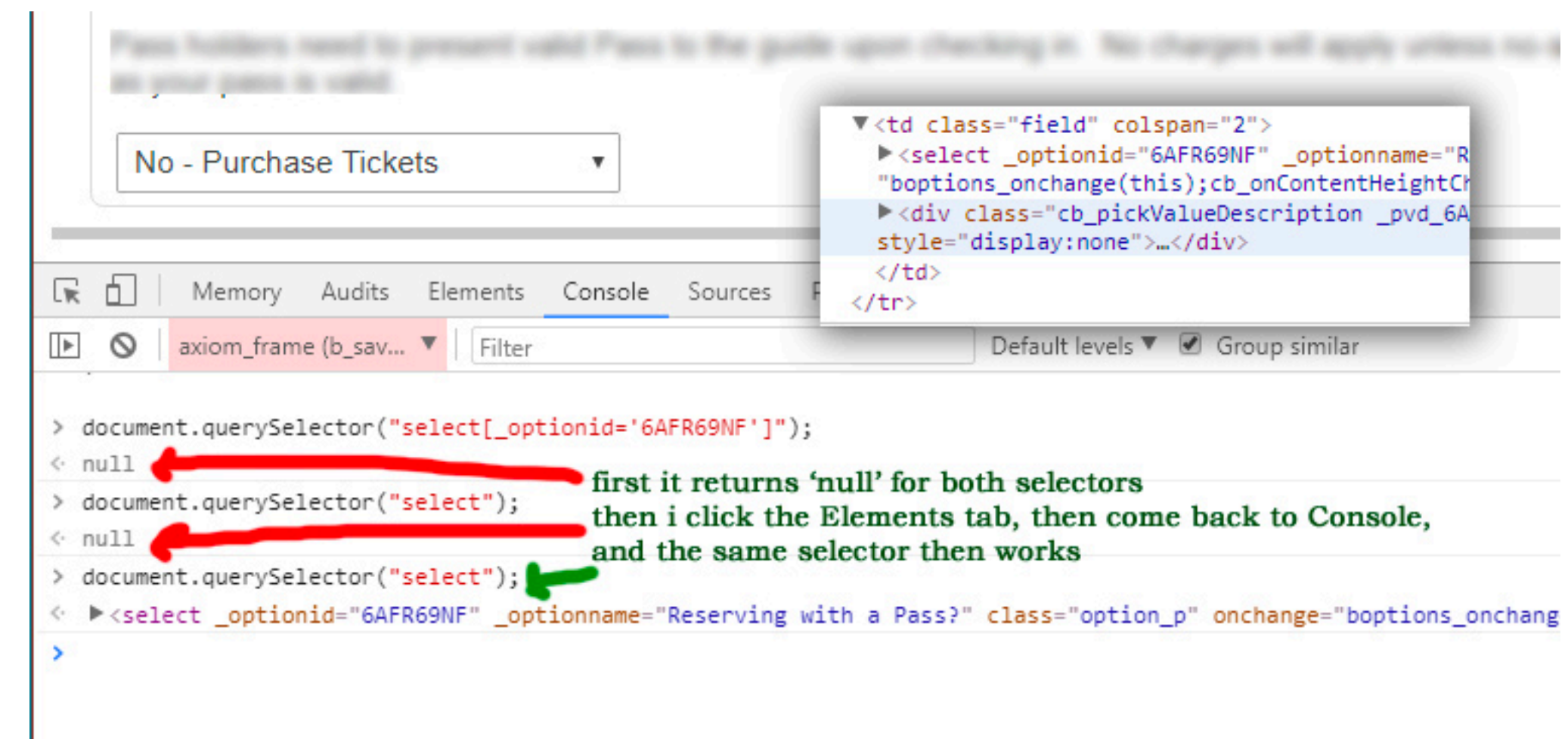
getAttribute / setAttribute

innerHTML / textContent

classList

style

addEventListener / removeEventListener



# Полезные ссылки

<https://www.javascripttutorial.net/javascript-dom/>

<https://habr.com/ru/company/ruvds/blog/416539/>