







# Frontender[1.0] JavaScript - DOM и CSS. Стили, классы, date-\* атрибуты

 YouTube	<a href="https://youtu.be/AJw9_euBO5E">https://youtu.be/AJw9_euBO5E</a>
 Telegram	<a href="https://t.me/Dmitry_Kolotilshikov">https://t.me/Dmitry_Kolotilshikov</a>
 Github	<a href="https://github.com/DmitryKolotilshikov/">https://github.com/DmitryKolotilshikov/</a>
 Boosty	<a href="https://boosty.to/dmitry_ko">https://boosty.to/dmitry_ko</a>
# Номер урока	30



Задачи к этому уроку тут [https://boosty.to/dmitry\\_ko](https://boosty.to/dmitry_ko)



Полезные ссылки:

<https://learn.javascript.ru/styles-and-classes>

<https://learn.javascript.ru/dom-attributes-and-properties>



- **Атрибуты** – это то, что написано в HTML.
- **Свойства** – это то, что находится в DOM-объектах.

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script src="script.js" defer></script>
  <link rel="stylesheet" href="styles.css">
  <title>DOM - styles, attributes</title>
</head>

<body>
  <div id="block-1">block</div>
  <button>click me</button>
```

```

    <div id="product" data-id="123#@%45" data-category="electronics" data-produce-year>
      Телевизор
    </div>
  </body>

</html>

```

```

.block {
  background-color: #295683;
  color: #fff;
  padding: 2em;
  border-radius: 10px;
  display: flex;
  justify-content: center;
  align-items: center;
  text-transform: uppercase;
  font-size: 2rem;
}

.block--xl {
  padding: 3em;
  font-size: 3rem;
}

```

```

// DOM - styles and attributes

const log = console.log;
const dir = console.dir;

document.body.style.backgroundColor = "lightgrey";

const block = document.querySelector("#block-1");

// block.style.backgroundColor = "#123456";
// block.style.color = "#fff";
// block.style.padding = "1em"; // следите за ед. измерения

// block.style.cssText = `
//   background-color: #123456;
//   color: #fff;
//   padding: 2em;
// `

// block.classList.add("block");
// log(block.classList.contains("block"));
// block.classList.remove("block");

// toggle - добавить класс, если его нет, иначе удалить.
// block.classList.toggle("block");
// block.classList.toggle("block");
// block.classList.toggle("block");

block.classList.add("block", "block--xl");

// 💎-----💎-----💎

```

```
/*  
как прочитать стиль?
```

Например, мы хотим знать размер, отступы, цвет элемента. Как это сделать?

Свойство `style` оперирует только значением атрибута `"style"`, без учёта CSS-каскада.

Что, если нам нужно, например, увеличить отступ на 20px?  
Для начала нужно его текущее значение получить.

Для этого есть метод: `getComputedStyle`  
\*/

```
const button = document.querySelector("button");  
log(button.style.color); // пусто  
log(button.style.backgroundColor); // пусто  
  
const buttonComputedStyles = getComputedStyle(button);  
  
dir(buttonComputedStyles);  
log(buttonComputedStyles.color);  
log(buttonComputedStyles.backgroundColor);  
  
button.style.fontSize = "16px";  
log(buttonComputedStyles.fontSize);  
  
log(buttonComputedStyles.paddingRight);  
log(buttonComputedStyles.paddingLeft);  
  
button.style.paddingRight = `${20 + parseInt(buttonComputedStyles.paddingRight)}px`;  
button.style.paddingLeft = `${20 + parseInt(buttonComputedStyles.paddingLeft)}px`;  
  
log(buttonComputedStyles.paddingRight);  
log(buttonComputedStyles.paddingLeft);  
  
// 💎-----💎-----💎  
/*  
getBoundingClientRect()  
возвращает размер элемента и его позицию относительно  
viewport (часть страницы, показанная на экране, и которую мы видим).  
*/  
log(button.getBoundingClientRect());  
log(buttonComputedStyles.height);  
log(buttonComputedStyles.width);  
  
// 💎-----💎-----💎  
// Атрибуты  
  
log(block.hasAttribute("id"))  
log(block.hasAttribute("class"));  
log(block.hasAttribute("href"));  
  
// log(block.getAttribute("class"));  
// log(block.removeAttribute("id"));  
  
block.setAttribute("title", "наш большой блок");
```

```

const link = document.createElement("a");
link.textContent = "ссылка на YouTube";
link.href = "https://youtube.com";
link.target = "_blank";

document.body.prepend(link);

// link.qwerty = "просто для теста";
link.setAttribute("qwerty", "просто для теста");

// 📡-----📡-----📡
// Data - Атрибуты
/*
В HTML атрибуты data-* позволяют хранить произвольные данные в
элементах.
Они помогают сохранять информацию непосредственно в
разметке, которую можно легко использовать через JavaScript.

Атрибуты data-* особенно полезны, когда нужно ассоциировать элемент
с дополнительными данными, которые не имеют стандартных HTML-атрибутов.
*/

const product = document.getElementById("product");

log(product.dataset);
log(product.dataset.id);
log(product.dataset.category);
log(product.dataset.produceYear);
product.dataset.produceYear = 1999;
log(product.dataset.produceYear);

/*
Преимущества использования data-* атрибутов:

- Гибкость: можно хранить любую информацию, связанную с элементом,
без необходимости вносить изменения в стандартную структуру HTML.

- Удобство интеграции с JavaScript: доступ к data-* атрибутам
осуществляется через объект dataset, что упрощает их использование.

- Совместимость: данные, хранящиеся в data-* атрибутах, не влияют на
стиль и поведение элементов по умолчанию.
*/

```