

# Стили

**№ урока:** 5    **Курс:** JavaScript Базовый

**Средства обучения:** Visual Studio Code  
Web Browser

## Обзор, цель и назначение урока

Научиться использовать CSS классы и редактировать отдельные стили элементов через JavaScript.

## Изучив материал данного занятия, учащийся сможет:

- Добавлять удалять CSS классы для элементов.
- Работать со стилями через свойство style.
- Использовать вычисляемые стили.

## Содержание урока

1. Работа со свойством className
2. Работа со свойством classList
3. Использование свойства style
4. Вычисленные стили и метод getComputedStyle

## Резюме

- CSS стили предпочтительный вариант оформления элементов на странице. Старайтесь избегать использования атрибута style и свойства style в JavaScript коде, через которые можно менять значения отдельных CSS свойств.  
Использование классов вместо отдельных стилей дает преимущество в производительности и простоте дальнейшего использования. Оформление сосредоточено в CSS файле, поэтому редактирование внешнего вида затрагивает только редактирование данного файла, а не разбор логики JavaScript приложения.
- Через JavaScript можно изменить как отдельный стиль, так и класс, который применяется для элемента.
- С помощью свойства className можно прочитать или поменять классы, которые используются элементом. Данное свойство дает доступ к атрибуту class элемента. Если присвоить свойству значение, то существующие классы, указанные в атрибуте class, будут перезаписаны.
- Свойство classList дает более гибкий интерфейс для работы с CSS классами элемента. Данное свойство является объектом, который предоставляет метод для добавления, удаления, проверке наличия класса в элементе.  
add – добавить класс, к уже имеющимся  
remove – удалить класс  
toggle – добавить, если класса нет, удалить, если класс есть

contains – проверка наличия указанного класса в списке классов

Также classList можно перебрать с помощью цикла for ... of.

- Для того чтобы работать с отдельными CSS свойствами элемента используется свойство style. Свойство представляет объект, который соответствует значениям, указанным в атрибуте style HTML элементам.

Имена свойств в объекте style соответствуют именам CSS свойств. Если в CSS свойство содержит дефис, то в JavaScript для этого свойства используется camel casing. Если свойство использует префикс, например, -moz или -webkit - такое свойство именуется с заглавной буквы.

CSS свойства	Свойство в JavaScript
background-color	backgroundColor
font-family	fontFamily
-moz-radial-gradient	MozRadialGradient

- Вычисленные стили – стили, полученные элементом, после применения всех CSS правил и разрешения всех геометрических свойств – высоты, ширины и т.д.

Для получения вычисленных стилей используется метод getComputedStyle(элемент, псевдоэлемент).

Например, если нужно узнать ширину элемента, для которого явно не было указано значение свойства width, чтение свойства style.width не даст результата. Так как ширину элемента вычислял браузер, и чтобы узнать текущий размер необходимо использовать метод getComputedStyle.

### Закрепление материала

- Почему лучше использовать CSS классы, а не отдельные CSS стили?
- В чем разница между свойствами className и classList?
- Как имена CSS свойств, в которых есть дефисы, используются совместно с свойством style?
- Как написать код, который добавит CSS класс, если в элементе такого класса нет, или удалит CSS класс, если в элементе такой класс уже есть?
- Что такое вычисленные стили и как их получить?

### Самостоятельная деятельность учащегося

Выполните задания в директории Exercises\Tasks\005 Styles. Текст задач находится в комментариях в task script.

### Рекомендуемые ресурсы

Style, работа с инлайновыми стилями

<https://developer.mozilla.org/ru/docs/Web/API/HTMLElement/style>

className

<https://developer.mozilla.org/ru/docs/Web/API/Element/className>

classList

<https://developer.mozilla.org/ru/docs/Web/API/Element/classList>

getComputedStyle

<https://developer.mozilla.org/ru/docs/Web/API/Window/getComputedStyle>