**ECMAScript** — это встраиваемый расширяемый не имеющий средств [ввода-вывода](https://ru.wikipedia.org/wiki/%D0%92%D0%B2%D0%BE%D0%B4-%D0%B2%D1%8B%D0%B2%D0%BE%D0%B4) язык программирования, используемый в качестве основы для построения других скриптовых языков[[3]](https://ru.wikipedia.org/wiki/ECMAScript#cite_note-ecmaZakas-4). Стандартизирован международной организацией [ECMA](https://ru.wikipedia.org/wiki/ECMA) в [спецификации](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F) **ECMA-262**.

Каждое издание ECMAScript получает аббревиатуру ES с последующим его номером. Всего существует 8 версий ECMAScript.

Версия ES6/ES2015 вышла в июне 2015 года. С выходом этого пакета обновлений комитет принял решение перейти к ежегодным обновлениям. Поэтому издание было переименовано в ES2015, чтобы отражать год релиза.

* добавлено [деструктурирующее присваивание](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment);
* добавлены [стрелочные функции](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Functions/Arrow_functions);
* в [шаблонных строках](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/template_strings) можно объявлять строки с помощью **`** (обратных кавычек). Шаблонные строки могут быть многострочными, также могут интерполироваться;
* **let** и **const** — альтернативы var для объявления переменных. Добавлена «временная мертвая зона»;
* итератор и протокол итерации теперь определяют способ перебора любого объекта, а не только массивов. **Symbol** используется для присвоения итератора к любому объекту;
* добавлены функции-генераторы. Они используют **yield** для создания последовательности элементов. Функции-генераторы могут использовать **yield\*** для делегирования в другую функцию генератора, кроме этого они могут возвращать объект генератора, который реализует оба протокола;
* добавлены промисы.

ES2016 (ES7) вышла в июне 2016 года. Среди изменений в этой версии ECMAScript можно отметить:

* оператор возведения в степень **\*\*;**
* **метод Array.prototype.includes**, который проверяет, содержится ли переданный аргумент в массиве.

Спустя еще год выходит версия ES2017 (ES8). Данный стандарт получил следующие изменения:

* асинхронность теперь официально поддерживается (**async/await**);
* «висячие» запятые в параметрах функций. Добавлена возможность ставить запятые в конце списка аргументов функций;
* добавлено два новых метода для работы со строками: **padStart()** и **padEnd()**. Метод **padStart()** подставляет дополнительные символы слева, перед началом строки. А **padEnd(),** в свою очередь, справа, после конца строки;
* добавлена функция **Object.getOwnPropertyDescriptors(),** которая возвращает массив с дескрипторами всех собственных свойств объекта;
* добавлено разделение памяти и объект **Atomics**.

Что же касается ES.Next, то этот термин является динамическим и автоматически ссылается на новую версию ECMAScript. Стоит отметить, что каждая новая версия приносит новые функции для языка.

## Выводы

* ECMAscript выходит ежегодно;
* первые пакеты обновления назывались ES1, ES2, ES3, ES4, ES5;
* новые выпуски (начиная с 2015 года) получили название ES2015, ES2016, ES2017 (аббревиатура ES + год выпуска);
* ECMAScript является стандартом, а JavaScript — это самая популярная реализация этого стандарта. Среди других реализаций можно отметить [SpiderMonkey](https://ru.wikipedia.org/wiki/SpiderMonkey" \t "_blank), [V8](https://ru.wikipedia.org/wiki/V8_(движок_JavaScript)) и [ActionScript](https://ru.wikipedia.org/wiki/ActionScript" \t "_blank).