

## Pesquisa sobre JavaScript e TypeScript

A linguagem JavaScript (JS) surgiu em 1995, teve seu primeiro protótipo criado em apenas 10 dias sendo batizada com o nome Mocha e, quando teve seu primeiro lançamento oficial, juntamente com a versão 2.0 do navegador Netscape, foi chamada de LiveScript. Nesta época a empresa Netscape estava em negociação com a empresa Sun Microsystems tendo a ideia era desenvolver tecnologias para rodar nos navegadores, e a Sun estava preste a lançar um dos seus maiores projetos no mercado a linguagem Java, e todos diziam na época um jargão bem famoso “Java é a linguagem do futuro” as duas empresas observando esse sucesso todo, tiveram uma ideia para fazer uma grande jogada de marketing, e eles fizeram, mas gera confusão até hoje, a grande ideia das duas foram juntar os nomes dos seus respectivos projetos LiveScript e Java, assim surgiu o nome JavaScript.

A linguagem estava em franco crescimento, para a sua evolução se mostrou necessário obedecer determinados padrões e normativas, os criadores do JS se associaram ao ECMA (European Computer Manufactures Association) em 1996. Como o nome Javascript já havia sido patenteado pela Sun Microsystems (atual Oracle), optou-se por se definir um novo nome à linguagem utilizando a junção das palavras ECMA e Javascript, surgindo então o ECMAScript.

Após essa sua padronização, o ECMAScript teve diversas versões, seguem elas em ordem cronológica:

1. **ECMAScript 1** – Lançada em 1997, a primeira versão da linguagem, mas com padrões e normativas definido após ECMA. Conhecida também como versão dos dez dias, pois foi desenvolvida neste período de tempo.
2. **ECMAScript 2** - Criada em agosto de 1998 para se adequar à ISO/IEC 16262.
3. **ECMAScript 3** – Dezembro de 1999, foi nessa versão onde a linguagem ganhou implementações importantes como *do-while*, expressões regulares, novos métodos para o objeto *string* além de tratamentos das exceções.
4. **ECMAScript 4** – concluída em 2008, esta versão teve seu desenvolvimento baseado em ML (Meta Language), uma linguagem de programação funcional mais utilizada em ambientes de pesquisa acadêmica. Devido à grande quantidade de mudanças em relação ao ECMAScript 3, o que tornaria esta versão totalmente disruptiva em relação às anteriores, a mesma foi abandonada pelo comitê técnico da ECMA-262, optando-se por se dar continuidade evolutiva à versão anterior.
5. **ECMAScript 5** – Após o impasse gerado com a versão 4, foi lançada em 2012 oficialmente a versão 5 do ECMAScript possuindo recursos valiosos como o suporte a JSON, métodos mais avançados de manipulação de arrays, getters e setters.

6. **ECMAScript 6** – Lançada em 2015, atribui recursos avançados à linguagem como reflection, collections, binary data.
7. **ECMAScript 7** – Também conhecida como ECMAScript 2016 (ano da sua conclusão), possui como operadores exponenciais, dentre outros;

Com a evolução da linguagem, logicamente algumas ferramentas são implementadas bem como aperfeiçoadas, e se tratando de uma linguagem que tem origem no JS, onde que decolou nos últimos anos em relação a sua utilização pela comunidade/mercado. Assim veremos algumas diferenças entre o ES5 e ES6:

- Suporte para constantes;
- Parâmetro de funções (*default parameters, rest parameters*)
- Programação Funcional (*arrow functions, destructing*)
- Classes
- Módulos
- Template String

Com TypeScript tem-se recursos que melhor suportam o uso da Programação Orientada a Objetos, que tem como base quatro princípios fundamentais: encapsulamento, herança, abstração e polimorfismo. A POO sempre foi um problema ao ser aplicada em JavaScript, devido a sua sintaxe não permitir escrever classes, por exemplo, de forma tão clara, além da fraca tipagem de dados. O TypeScript oferece então uma forma de corrigir ou contornar esses problemas, adicionando funcionalidades que quando compiladas resultarão em código JavaScript novamente. Porém, agora o desenvolvedor lidará diretamente com uma sintaxe simplificada, mais clara e amplamente suportada por editores de código modernos. Baseado nas especificações do ES 6, o TypeScript é capaz de oferecer recursos como interfaces e generics, características já presentes em outras linguagens de programação orientadas a objetos, como C# e Java.

Link para o repositório:

<https://github.com/TiagoBehenck/pesquisa-es5-es6>

Links Utilizados para fazer a pesquisa

<https://medium.com/@matheusml/o-guia-do-es6-tudo-que-voc%C3%AA-precisa-saber-8c287876325f>

[https://www.w3schools.com/js/js\\_es5.asp](https://www.w3schools.com/js/js_es5.asp)

[https://www.w3schools.com/js/js\\_es6.asp](https://www.w3schools.com/js/js_es6.asp)

<http://shipit.resultadosdigitais.com.br/blog/javascript-1-uma-breve-historia-da-linguagem/>

<https://medium.com/trainingcenter/afinal-javascript-e-ecmascript-s%C3%A3o-a-mesma-coisa-498374abbc47>

<https://medium.com/iclinic/es6-es2015-o-que-mudou-c22d9308f52d>

<https://www.devmedia.com.br/introducao-ao-typescript/36729>

<http://www.ecma-international.org/publications/standards/Ecma-262.htm>