Saulo Carvalho

**Instruções do projeto**

Sabemos que a evolução da linguagem JavaScript para ECMAScript proporcionou algumas mudanças na forma de programar para a weeb. Diante disso, pesquise as principais vantagens dessa evolução através das especificações do ECMAScript (ECMA).

**Resolução**

O principal objetivo da nova versão especificação foi tornar a linguagem mais flexível, enxuta e fácil de se aprender e trabalhar, tornando-a mais próxima a outras linguagens orientadas a objeto, como Java e Python.

Dentre as principais mudanças, temos:

* Criação de novos tipos de dados (Map, WeakMap, Set, WeakSet);
* Novas maneiras de iterar objetos e coleções;
* Declaração de variáveis com let e const;
* Modularização e estrutura de classes;
* Geradores e símbolos;
* Operadores rest e spread.

Nos últimos anos, o JavaScript ganhou muita força na comunidade de desenvolvedores através do “grande boom” do ecossistema das tecnologias Node.js e npm. Desde então, a flexibilidade da linguagem tem sido usada a favor não somente do lado do cliente (nos navegadores), mas também do lado do servidor, algo impensável até alguns anos atrás. Tudo isso fez com que não somente a comunidade, mas também grandes empresas de tecnologia como o Facebook, reconhecessem a força da linguagem e finalmente a levasse a sério.

E as evoluções não param por aí. Para que a linguagem não pare novamente no tempo, a Ecma Internacional se comprometeu a lançar uma nova versão da especificação todos os anos. Desde o ano passado, a sétima edição da especificação está sendo elaborada e promete trazer ainda mais melhorias, tais como:

* Novas operações em Array;
* Operador de exponenciação (\*\*);
* Decoradores;
* Propriedades estáticas de classe;
* Async-await.

Referência:

<https://imasters.com.br/front-end/o-ecmascript-6-e-o-futuro-do-javascript>