**ECMASCRIPT**

1. **O que é?**

Para um melhor entendimento, é necessário saber sobre **a origem do JavaScript**

- O JS for originalmente desenvolvido por Brendan Eich em 1995, para ser utilizado no Navigator2

- Rapidamente JavaScript ganhou popularidade e aceitação como linguagem de script, cliente side de páginas web

- Chamou a atenção da Microsoft que desenvolveu uma implementação da linguagem, que foi chamada de JScript para evitar problemas de licenciamento. O JScript foi inserido na internet em 1996

- Por volta de 1997, foi anunciado que o JavaScript seria enviado para uma equipe internacional, a Associação Européia dos Fabricantes de Computadores(ECMA), com o intuito de fazer do Javascript uma linguagem de padrão industrial

- A ECMA publicou então, o padrão 262, se popularizando como ECMAScript

- O ECMAScript é uma especificação, sendo assim, não está ligado diretamente aos navegadores, a ideia é aumentar a compatibilidade das aplicações em browsers diferentes

- A primeira versão foi disponibilizada em 1997, desde então, foram surgindo mais versões, sendo a mais significativa, a versão 5, padronizada em 2009, suportada de forma quase completa em browsers modernos, em junho de 2015, foi publicada a versão 6, ES6 ou ECMAScript 2015

1. **Babel**

**-** Um dos compiladores mais completos em termos de transcrição de código escrito em ES6 para ES5

- Com ele, os códigos escritos em ES6 serão transcritos para o ES6, versão suportada pelos browsers em 2018, porém, atualmente os browsers já suportam as features ES6, portanto, a configuração do Babel não será mais necessária

<https://babeljs.io/> > Setup > In the browser > babel-standalone > copiar cod js

1. **Var e Let**

A diferença entre eles, é que o Let preserva o escopo aonde a variável foi criada, ou seja, só pode ser utilizada depois do bloco de código que foi criada; diferentemente de variáveis declaradas com var podem ser utilizadas antes de sua declaração

São três os tipos de escopos: global, função e bloco

Em termos de escopo de bloco variáveis declaradas como var podem ser utilizadas dentro de funções e dentro de blocos

Variáveis, entre outras coisas declaradas no escopo global, fica disponível para tudo, o global é basicamente o objeto window

1. **Introdução a Orientação a Objetos – Considerações preliminares**

***Paradigmas de programação***

Paradigmas – Padrão; conduta; uma forma de se fazer alguma coisa

***Procedural/Estruturado/Imperativo***– Funcional/lógico

switch{}, if(){}else{}

***Orientado a Objetos***– Concorrente/reativo

While, do while

***Paradigma Procedural***

- Chamada de procedimentos (procedures) para manipulação de dados

***Paradigma de Orientação a Objetos***

- Estruturas de dados com comportamentos

Atributos são declarados dentro de objetos

Var > classe > objeto > atributo

**Pilares da orientação a objeto** – abstração, encapsulamento, herança, polimorfismo

***Abstração*** – **Entidade :** compreensão de que um objeto do mundo real deve passar para dentro da aplicação; **Identidade :**  identificação única da identidade; **Características :** atributos que especificam as características da entidade; **Ações :** métodos de comportamentos dos objetos na vida real, que fazem sentido no contexto da aplicação.

- Habilidade intuitiva de conhecimento das coisas

- Compreende coisas do mundo real e trás para o código

- As ações e características tem a necessidade de serem dosadas, para que que faça sentido na aplicação