HISTÓRIA E CONCEITOS

**História:**

• Lançado em setembro de 1995

• Criada por Brendan Eich

• ECMAScript (Padrões)

• TC39 (Grupos)

• Fluxo das propostas

**Stage 0: strawman** (submissão de formulário com a ideia para submeter o script. Documento revisado)

**Stage 1: proposal** (Proposta formal da funcionalidade. Membro TC39. Objetivos e exemplos. Possíveis alterações)

**Stage 2: draft** (Proposta com sintaxe e semântica da funcionalidade. Alterações incrementais)

**Stage 3: candidate** (Quase finalizada. Feedback de implementação. 2 implementações compatíveis com a ECMA)

**Stage 4: finished** (Proposta madura.Passada na suit de testes. Editor ECMA deve assinar a especificação)

• ES2018

Operações rest/spread

Iterações assíncrona

Promise.prototype.finally()

• ES.Next

Futuras implementações

Disponivel em Babeljs.io

**Conceito:**

• Linguagem interpretada.

• Tipagem fraca e dinâmica.

• Typescript e boas práticas (Superset):

<https://www.typescriptlang.org>

Interfaces, Classes, Tipagem nas variáveis, entre outros.

• Flow

<https://www.flow.org/en>

• Funções de primeira classe e de ordem maior

- A função pode ser atribuída a uma variável

- Atribuída a uma estrutura de dados (object ou array)

- Pode ser passada por argumentos

- Também pode ser retornada por outras funções

function getName(){

    return "Jacques de Lassau A. Cheron";

}

function logFn(fn){

    console.log(fn);

}

const logFnResult = logFn;

const obj = {

    logFn: logFn

}

const arr = [logFn];

logFnResult(getName);

• Closure

- É a capacidade de uma função lembrar do ambiente em que foi criada, com relação a escopo.

Global, de função e bloco.

    function init(){

        const exemplo = `Essa variável`;

        return function(){

            console.log(`1 - O valor da variável exemplo é: ${exemplo}`);

            return function(){

                console.log(`2 - O valor da variável exemplo é: ${exemplo}`);

                return function(){

                    console.log(`1 - O valor da variável exemplo é: ${exemplo}`);

                }

            }

        }

    }

    const initFn1 = init();

    const initFn2 = initFn1();

    const initFn3 = initFn2();

    initFn3();