

//FUNÇÕES CONSTRUTORAS:

//Funções construtoras são funções no Javascript que são utilizadas como se fossem "Classes" em Java, pois por através delas é possível criar funções que são capazes de armazenar atributos e métodos, gerar encapsulamento e polimorfismo e o principal, elas podem ser instanciadas normalmente, como acontece no Java.

//OBS: Existem sim classes em Javascript, porém, no final das contas elas não passam de funções se fingindo de classes. Elas foram criadas com o intuito de não serem um paradigma muito chocante para desenvolvedores que já vêm de linguagens que utilizam classes de maneira tradicional.

//FUNÇÃO CONSTRUTORA SIMPLES:

```
function Carro(velocidadeMaxima = 200, delta = 5) { //Criamos uma função Construtora que tem o objetivo de servir como uma classe para o objeto carro, sua função é medir a aceleração de um carro a cada pisada de acordo com a velocidade delta dele...
```

//Atributo privado: qualquer atributo criado com var, let ou const dentro de uma função construtora não pode ser acessado por fora, isso faz com que qualquer variável ou função criada dentro de uma função construtora não possa ser acessado a partir do seu objeto. Para ele ser acessado, temos que usar o "this".

```
  var velocidadeAtual = 0;
```

//Método Público: por usarmos o "this" conseguimos acessar o método a partir de fora da função construtora.

```
this.acelerar = () => { //Somente quando colocamos o "this" é que temos de fato uma função construtora...
```

```
  if (velocidadeAtual + delta <= velocidadeMaxima) {
```

```
    velocidadeAtual += delta;
```

```
  } else {
```

```
    velocidadeAtual = velocidadeMaxima;
```

```
  }
```

```
}
```

//Uma espécie de método getter para o método privado "velocidadeAtual":

```
this.getVelocidadeAtual = () => velocidadeAtual; //Perceba que por usar o "this" sobre uma função que tem o objetivo apenas mostrar a velocidade atual de um carro, conseguimos criar uma espécie de getter para mostrar o atributo privado...
```

```
}
```

```
console.log("1) Velocidades dos carros...")
```

```
const uno = new Carro; //Criamos um objeto instanciando a função construtora "Carro" (Não passamos nenhum parâmetro por que o uno já atende o padrão)...
```

```
uno.acelerar();
```

```
uno.acelerar();
uno.acelerar(); //... aceleramos o uno 3 vezes por através do seu nome de objeto e o seu método público...
console.log(`Uno depois de 3 aceleradas está á: ${uno.getVelocidadeAtual()}Km/H`); //... No fim simplesmente mostramos a velocidade e m que o uno está depois de 3 aceleradas no carro...

const ferrari = new Carro(350, 20); //Veja que com a ferrari já mudamos os parâmetro para uma velocidade maior...
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar();
ferrari.acelerar(); //... aceleramos mais vezes, a cada acelerada ela pega 20km de velocidade...
console.log(`Ferrari depois de acelerar 9 vezes está á: ${ferrari.getVelocidadeAtual()}Km/H`); //... No fim simplesmente mostramos a velocidade em que o ferrari está depois de 9 aceleradas no carro...
```

RESULTADO NO CONSOLE...

```
[Running] node "c:\Users\User\Documents\JAVASCRIPT\ARQUIVOS_DAS_AULAS\070-Funcoes_Construtoras.js"
1) Velocidades dos carros...
Uno depois de 3 aceleradas está á: 15Km/H
Ferrari depois de acelerar 9 vezes está á: 180Km/H

[Done] exited with code=0 in 0.2 seconds
```