

```
//CONSTRUCTORS:
```

//Já vimos em funções que construtores são úteis para quando desejamos criar objetos e só acessar seus atributos e métodos dentro da função construtora.

//Eles também são úteis para organização do nosso código, para que possamos receber todos os atributos diretamente por parâmetro.

//Além disso o construtor quando usamos o "this" faz com que os atributos recebidos só possam ser acessados por através do "this" fora do método construtor.

```
class Client {
  constructor(nome, cpf){
    this.nome = nome //Recebemos os atributos com o "this"
    this._cpf = cpf
  }

  verNomeComThis(){
    console.log(this.nome) //Aqui temos um método que usa "this"
  }

  verNomeSemThis(){
    console.log(nome) //E temos um método que não usa "this", por isso não conseguirá acessar aos atributos...
  }
}
```

```
let cliente_0001 = new Client('Fabiano', '123.456.789-1')
console.log(cliente_0001)
/*RESULTADO NO CONSOLE:
Client { nome: 'Fabiano', _cpf: '123.456.789-1' }
*/
```

```
cliente_0001.verNomeComThis()
/*RESULTADO NO CONSOLE:
Fabiano
*/
```

```
cliente_0001.verNomeSemThis()
/*RESULTADO NO CONSOLE:
file:///c:/Users/Almoxarifado/Documents/javascript/arquivos_das_aulas/186-
constructor___Entendendo_a_utilidade_de_construtores_dentro_de_uma_classe.js:18
    console.log(nome)
            ^

ReferenceError: nome is not defined
*/
```