

```
//POLIMORFISMO:
```

//Como já deveríamos saber, o Polimorfismo é um conceito de POO onde podemos modificar os métodos de uma classe filha á partir dos métodos já existentes na sua classe mãe, para que eles se adequem melhor a classe filha. Esse conceito também pode ser reconhecido como "Especialização".

```
//Para isso, dentro do Polimorfismo vamos trabalhar com 2 regras básicas que são:
```

```
/*      # POLIMORFISMO DE SOBRECARGA: Quando mudamos a assinatura do método (acrescentando parâmetros por exemplo);  
      # POLIMORFISMO DE SOBRESCRITA: Quando mudamos o comportamento de um método dentro do seu bloco de execução;  
*/
```

```
//EXEMPLO DE POLIMORFISMO DE SOBRECARGA E SOBRESCRITA:
```

```
class Pessoa { //Temos uma classe pessoa que recebe somente 2 parâmetros...
```

```
  constructor(nome, idade){  
    this.nome = nome  
    this.idade = idade  
  }
```

```
  mostraTempo(){ //Além disso tem um método que não recebe parâmetro nenhum e como execução somente mostra a idade da  
    console.log(`A idade de ${this.nome} é ${this.idade}.`)    //pessoa...  
  }
```

```
}
```

```
class Profissao extends Pessoa { //E temos uma classe "Profissao" que estende de Pessoa, mas tem uma sobrecarga no
```

```
  constructor(nome, idade, profissao){ //constructor, onde também receberá a profissão da pessoa...  
    super(nome, idade)  
    this.profissao = profissao  
  }
```

```
  mostraTempo(anos){ //Note que temos uma sobrescrita no método "mostraTempo", ele recebe parâmetros agora e além  
    console.log(`Eu ${this.nome} trabalho de ${this.profissao} a ${anos} anos.`) //disso possuí um bloco de execução  
  }                                     //totalmente diferente!
```

```
}
```

```
let ana = new Pessoa('Ana', '21') //Note que na instância de ana o método mostra tempo nos retorna só a idade dela...
```

```
ana.mostraTempo()
```

```
/*RESULTADO NO CONSOLE:  
A idade de Ana é 21.  
*/  
  
let ricardo = new Profissao('Ricardo', '31', 'Pedreiro') //Mas no Ricardo, o mostra tempo tem que receber um parâmetro,  
ricardo.mostraTempo(10) //e além disso, tem resultado diferente...  
/*RESULTADO NO CONSOLE:  
Eu Ricardo trabalho de Pedreiro a 10 anos.  
*/  
  
//Perceba que a sobrecarga é usada para quando desejamos incrementar dados a uma instância!  
//E usamos a sobrescrita quando desejamos que um método da classe mãe se comporte de forma diferente na classe filha!
```