



# CC4670

# COMPUTAÇÃO MÓVEL

Prof. Isaac Jesus  
Prof. Rafael Alves



PRÁTICA

# JavaScript

JS

ES

1. Refatore o código a seguir para que declaração do objeto VideoGame seja uma classe de acordo com o ES6. Não se esqueça de criar o construtor e invocar um objeto desta classe

```
function VideoGame(marca, nControles, tipoMidia) {  
  this.marca = marca;  
  this.nControles = nControles;  
  this.tipoMidia = tipoMidia;  
}  
  
var playstation = new VideoGame('sony', '2', 'dvd');  
console.log(playstation);  
// { marca: 'sony', nControles: '2', tipoMidia: 'dvd' }
```

# JavaScript

JS

ES

1. Inclua o atributo *ligado*

Inclua os métodos: *jogar()*, *ligar(estado)* e *salvarJogo()*;

# JavaScript

JS ES

2. Crie duas subclasses (**Medico** e **Enfermeira**) a partir de uma superclasse chamada **FuncionariosDoHospital**. Abaixo, listamos as propriedades e métodos que você encontrará nas classes Medico e Enfermeira.

- a. Medico:

Propriedades: **\_nome**, **\_numeroRestantesDeFerias** (= 20 dentro de constructor()), **\_cpf**

Methods: **tirarFerias(num\_dias)**

# JavaScript

JS ES

2. Crie duas subclasses (**Medico** e **Enfermeira**) a partir de uma superclasse chamada **FuncionariosDoHospital**. Abaixo, listamos as propriedades e métodos que você encontrará nas classes Medico e Enfermeira.

- a. Enfermeira:

Propriedades: ***\_nome***, ***\_numeroRestantesDeFerias*** (= 20 dentro de constructor()),  
***\_certificados***

Methods: ***tirarFerias(num\_dias)***, ***adicionarCertificado(certificado)***

3. Declare uma função comum. Essa função deverá possuir dois parâmetros (nome, sobrenome). A função deve mostrar no console a seguinte frase: *"Olá <nome> <sobrenome>, eu sou uma função!"*.

Depois você deverá:

Replicar o exercício, utilizando agora uma **Arrow Function**. Utilize os mesmos parâmetros e retorne no console a mesma frase.

# JavaScript

JS ES

4. Considere a seguinte função já implementada:

```
const processarMensagem = (mensagem, callback) => callback(mensagem);
```

Crie arrow functions diferentes como callback para que, ao passá-las para a função **processarMensagem**, sejam produzidas exatamente as seguintes saídas:

- a) **Junta duas strings:** Saída: \*\*\* Atenção: Sistema instável! \*\*\*
- b) **Calcula o número de caracteres:** Saída: Resumo: A mensagem contém 42 caracteres.
- c) **Informa se a mensagem é composta por somente maiúscula ou minúscula, ou ambas:**  
Saída: Mensagem em caixa baixa: tudo em minúsculas



# JavaScript

JS ES

5. Implemente o diagrama abaixo, teste depois de implementado.

