

# Desafios Algoritmos / JavaScript

## 1) Verificar elegibilidade para votar

Crie uma função que verifique se uma pessoa é elegível para votar.

A pessoa precisa ter 18 anos ou mais para votar, além disso, vamos incluir uma verificação para ver se a pessoa tem menos de 75 anos, assumindo que acima dessa idade, o voto é facultativo.

- Com base na idade fornecida deverá retornar "Elegível para votar", caso seja maior que 75 retornar "Voto facultativo" ou caso seja menor de 18 retornar "Não elegível para votar"

## 2) Verificar se todos os números são positivos

Este desafio recebe três números como parametro da função e verifica se todos são positivos.

Se todos forem positivos, retorna verdadeiro; caso contrário, retorna falso.

## 3) Classificar uma temperatura

Este desafio classifica a temperatura em 'Frio', 'Moderado' ou 'Quente'.

A temperatura é considerada 'Fria' se for menor que 10°C, 'Moderada' entre 10°C e 25°C, e 'Quente' se for maior que 25°C.

Crie uma função que vai verificar a temperatura enviada e retornar se está 'Frio', 'Moderado' ou 'Quente' com base na temperatura enviada.

## 4) Algoritmo para classificar numero par ou ímpar

Escreva uma função que receba um número e retorne "Par" se o número for par e "Ímpar" se o número for ímpar.

### **5) Classificar pessoas maiores de idade**

Dado um array de objetos onde cada objeto tem as propriedades nome e idade, escreva uma função que retorne um novo array contendo apenas as pessoas com 18 anos ou mais.

### **6) Imprima na tela todos os números de 1 a 10.**

### **7) Imprima na tela a tabuada de multiplicação de 5.**

### **8) Dado um número de 1 a 7, imprima o dia da semana correspondente.**

Entendendo que o 1 dia da semana é domingo e o ultimo dia o 7 é o sabado, deve criar um algoritmo quando usuario pesquisar entre 1 ao 7 dia deverá retornar o dia da semana respectivo.

### **9) Crie uma função chamada adicionarElemento**

Crie uma função chamada adicionarElemento que aceite dois parâmetros: um array e um elemento.

- A função deverá adicionar o elemento recebido como parametro dentro do array porem precisa verificar se esse elemento que deseja adicionar já não exista no array, caso ele já exista dentro do array não deverá ser adicionado ao array.

- Ao final o array não deve ter itens duplicado se tentar chamar a função duas vezes tentando adicionar mais de uma vez o mesmo elemento no array.

## 10) Você tem essa lista abaixo de contatos:

```
let listaDeContatos = [  
  { nome: "João", telefone: "123456789" },  
  { nome: "Maria", telefone: "987654321" },  
  { nome: "Matheus", telefone: "111222333" },  
  { nome: "Matheus Fraga", telefone: "444555666" },  
  { nome: "Ana", telefone: "999888777" }  
];
```

- Agora crie em seguida uma função que deverá listar todos os contatos da lista que tem o nome de Matheus.