

# Laboratório 2

## Sistemas Distribuídos

Prof Adailton de Jesus Cerqueira Junior

Neste laboratório vamos apresentar o conceito de *truthy* e *falsy* do JavaScript. Também faremos uma pequena prática utilizando vetores (arrays).

## Falsy e Truthy:

1 – Crie uma pasta chamada lab02. Nesta pasta, crie um arquivo chamado **comparacao** com a extensão **.js (comparacao.js)**. Este arquivo terá alguns prints em JavaScript com os valores *falsy* e *truthy*.

```
console.log("Testando valores falsy e truthy");
console.log(false, ` é "${Boolean(false)}".`);
console.log(0, ` é "${Boolean(0)}".`);
console.log("", ` é "${Boolean("")}"`);
console.log(null, ` é "${Boolean(null)}".`);
console.log(undefined, ` é "${Boolean(undefined)}".`);
console.log(NaN, ` é "${Boolean(NaN)}".`);
console.log(true, Boolean(true));
console.log(1, Boolean(1));
console.log("Olá", Boolean("Olá"));
console.log([], Boolean([]));
console.log({}, Boolean({}));
```

2 – Agora vamos realizar algumas comparações em JavaScript.

```
console.log("Testando comparações");
console.log(1 == 1);
console.log(1 == "1");
console.log(1 === 1);
console.log(1 === "1");
console.log(true == 1);
console.log(1 == [1]);
console.log(null == null);
console.log(null == undefined);
console.log([] == false);
console.log([] == []);
```

3 – Vamos testar operadores de comparação em JavaScript. Primeiro vamos comparar dois números inteiros e imprimir qual é a relação entre eles.

```

console.log("Testando operadores de comparação");
let num1 = 10;
let num2 = 5;

if (num1 > num2) {
    console.log(`${num1} é maior que ${num2}.`);
} else if (num1 < num2) {
    console.log(`${num1} é menor que ${num2}.`);
} else {
    console.log(`${num1} é igual a ${num2}.`);
}

```

4 – Vamos utilizar o operador lógico AND em JavaScript.

```

console.log("Testando operador lógico AND (&&)");
let idade = 25;
let possuiCNH = true;

if (idade >= 18 && possuiCNH) {
    console.log("Você pode dirigir.");
} else {
    console.log("Você não pode dirigir.");
}

```

5 – Vamos utilizar o operador lógico OR em JavaScript.

```

console.log("Testando operador lógico OR (||)");
let temDinheiro = false;
let temCartaoDeCredito = true;

if (temDinheiro || temCartaoDeCredito) {
    console.log("Você pode fazer a compra.");
} else {
    console.log("Você não pode fazer a compra.");
}

```

6 – Vamos utilizar o operador lógico OR em JavaScript.

```

console.log("Testando a combinação dos operadores lógicos AND e OR");
let estaChovendo = true;
let temGuardaChuva = false;
let temCapaDeChuva = false;

// Verifica se a pessoa pode sair de casa
if ((estaChovendo && (temGuardaChuva || temCapaDeChuva)) || !estaChovendo) {
    console.log("Você pode sair de casa.");
} else {
    console.log("Você não pode sair de casa.");
}

```

# Vetores:

1 – Na pasta lab02, crie um arquivo chamado **vetores** com a extensão **.js (vetores.js)**. Neste arquivo iremos trabalhar um pouco com um vetor numérico.

```
// Definindo um vetor de números
const numeros = [1, 2, 3, 4, 5];

// Acessando elementos do vetor
console.log('Primeiro elemento:', numeros[0]);
console.log('Terceiro elemento:', numeros[2]);

// Adicionando um elemento ao final do vetor
numeros.push(6);
console.log('Vetor após adicionar um elemento:', numeros);

// Removendo o último elemento do vetor
const ultimoElemento = numeros.pop();
console.log('Elemento removido:', ultimoElemento);
console.log('Vetor após remover o último elemento:', numeros);

// Iterando sobre os elementos do vetor
console.log('Iterando sobre os elementos do vetor:');
numeros.forEach((numero, index) => {
  console.log(`Elemento no índice ${index}: ${numero}`);
});
```

2 – Vamos testar outras funções em JavaScript para manipular um vetor.

```
// Definindo um vetor de filmes
const filmes = ['Inception', 'The Matrix', 'Interstellar', 'Lord of the Rings'];

// Adicionando um elemento ao início do vetor
filmes.unshift('The Godfather');
console.log('Vetor de filmes após adicionar um elemento ao início:', filmes);

// Removendo o primeiro elemento do vetor
const primeiro = filmes.shift();
console.log('Primeiro filme removido:', primeiro);
console.log('Vetor de filmes após remover o primeiro elemento:', filmes);

// Encontrando o índice de um elemento
const indice = filmes.indexOf('The Matrix');
console.log('Índice de The Matrix:', indice);

// Verificando se um elemento está no vetor
const temInception = filmes.includes('Inception');
console.log('O vetor de filmes contém Inception?', temInception);

// Iterando sobre os elementos do vetor
console.log('Iterando sobre os elementos do vetor:');
filmes.forEach((filme, index) => {
  console.log(`Elemento no índice ${index}: ${filme}`);
});
```