

INSTITUTO MAUÁ DE TECNOLOGIA



DESENVOLVIMENTO DE SITES RESPONSIVOS E MODERNOS

Javascript

JAVASCRIPT

Linguagem de programação de 'alto nível'

PRINCIPAIS TIPOS DE DADOS E LITERAIS

```
// String
var nome = 'Dobbin';

// Numérico
var idade = 28;

// Booleano
var isHumano = true;

// Objeto
var professor = {
  nome: 'Dobbin',
  idade: 28,
  isHumano: true
};

// Palavra-chave que indica valor nulo
null

// Propriedade superior cujo valor é indefinido
undefined

// Array (Literal)
```

```
var vogais = ['a', 'e', 'i', 'o', 'u'];  
var primos = [3, 5, 7, 13];  
var mix = [1, 'palavra', {a: 123, b: 456}]
```

MANIPULAR OBJETO

```
var professor = {  
  nome: 'Dobbin',  
  idade: 28,  
  isHumano: true  
};  
  
alert(professor.nome); // output 'Dobbin'  
  
alert(professor['nome']); // output 'Dobbin'  
  
professor.nome = 'Rodrigo';  
alert(professor.nome); // output 'Rodrigo'  
  
professor.sobrenome = 'Fellows';  
alert(professor.sobrenome); // output 'Fellows'
```

MANIPULAR ARRAY

```
var vogais = ['a', 'e', 'i', 'o', 'u'];  
alert(vogais[0]) // output 'a'  
vogais.length // 5  
vogais.push('z'); // ['a', 'e', 'i', 'o', 'u', 'z'];  
vogais.pop(); // ['a', 'e', 'i', 'o', 'u'];
```

VARIÁVEIS

```
var teste = 'teste';  
// Declara uma variável, opcionalmente, inicializando-a com um va  
  
if (10 >= 5) {  
    var x = 5;  
}  
console.log(x); // 5  
  
let teste = 'teste';  
// Declara uma variável local de escopo do bloco, opcionalmente,  
  
if (10 >= 5) {  
    let y = 5;  
}  
console.log(y); // ReferenceError: y não está definido  
  
const teste = 'teste';  
// Declara uma constante apenas de leitura.  
  
var funcaoA = function () {  
    alert('Isto é uma função');  
}
```

```
funcaoA(); // Isto é uma função
```

```
function funcaoB () {  
  alert('Isto também é uma função');  
}
```

```
funcaoB(); // Isto também é uma função
```


OPERADORES DE COMPARAÇÃO

```
const a = 10; // Numérico
const b = '10'; // String
const c = 25; // Numérico
const d = 10; // Numérico

// Igual ==
a == b // true

// Não igual !=
a != d // false

// Estritamente igual ===
a === b // false

// Estritamente não igual !==
a !== b // true

// Maior que >
c > b // true

// Maior que ou igual >=
d >= c // false
```

```
// Menor que <  
a < d // false
```

```
// Menor que ou igual <=  
a <= d // true
```

CONDICIONAL IF, ELSE

```
// Exemplo 1
```

```
if (a > b) {  
    alert('A maior que B');  
}
```

```
// Exemplo 2
```

```
if (a > b) {  
    alert('A maior que B');  
} else {  
    alert('B maior que A');  
}
```

```
// Exemplo 3
```

```
if (a > b) {  
    alert('A maior que B');  
} else if (c > b) {  
    alert('B maior que A e menor que C');  
} else {  
    alert('B maior que A e maior que C');  
}
```

SWITCH

```
switch (expressao) {  
  case 'verde':  
    alert('Cor verde');  
    break;  
  case 'amarela':  
    alert('Cor amarela');  
  case 'azul':  
    alert('Cor azul');  
    break;  
  case 'vermelha':  
    alert('Cor vermelha');  
    break;  
  default:  
    alert('Cor preta');  
    break;  
}
```

FOR LOOP

```
let vogais = ['a', 'e', 'i', 'o', 'u'];  
  
for (let indice = 0; indice < vogais.length; indice++) {  
    alert(vogais[indice]);  
}  
  
for (let indice of vogais) {  
    alert(indice);  
}
```

WHILE LOOP

```
let x = 0;  
while (x < 10) {  
  alert(x);  
  x++;  
}
```

```
let y = 0;  
do {  
  alert(y);  
  y++;  
} while (y < 10)
```

EXEMPLOS

PERGUNTAS?

REFERÊNCIAS

- Sintaxe e tipos -> https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Guide/Values,_variables,_and_literals
- Expressões e operadores -> https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Guide/Expressions_and_Operators

ATIVIDADE

- Aplicar dinamismo no projeto