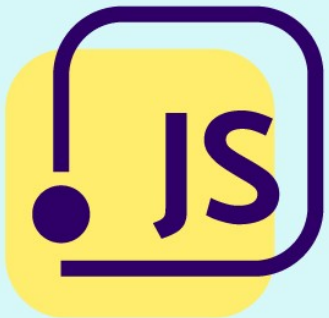


JavaScript

Prof. Daniel Di Domenico



Variáveis e Operadores

Variáveis

- Utilizar a palavra **var** para declarar as variáveis
 - Pode-se utilizar também **let** e **const**
- A tipagem é dinâmica funcionando de acordo com o valor atribuído

```
var x = 3; //variável int  
var y = "algoritmo"; //variável string  
var z = 1.8; //variável float  
var w = true; //variável boolean
```

```
//Checagem de tipo  
console.log(typeof(y)); //Imprime string
```

Operadores matemáticos

- Devem ser utilizados com o tipos numéricos

```
var x = 8;
```

```
var y = 5;
```

```
var a = x + y; //Resultado: 13
```

```
var b = x - y; //Resultado: 3
```

```
var c = x * y; //Resultado: 40
```

```
var d = x / y; //Resultado: 1.6
```

```
var d = x % y; //Resultado: 3 (operador de resto da divisão)
```

Operadores relacionais

- Comparações que resultam em Verdadeiro (**true**) ou Falso (**false**)

```
var x = 8;
```

```
var y = 5;
```

```
var a = x > y; //Resultado: true
```

```
var b = x < y; //Resultado: false
```

```
var c = x == y; //Igualdade – Resultado: false
```

```
var d = x != y; //Diferente - Resultado: true
```

```
var e = x >= y; //Resultado: true
```

```
var f = x <= y; //Resultado: false
```

Operadores lógicos

- Comparações que resultam em Verdadeiro (**true**) ou Falso (**false**)

```
var x = true;
```

```
var y = false;
```

```
var a = x && y; //E – Resultado: false
```

```
var b = x || y; //OU - Resultado: true
```

```
var c = ! x; //NOT (NÃO) – Resultado: false
```

Strings

- Utilizados para armazenar valores literais
 - São identificados pelo operador " " ou ' '
 - Possui propriedades e métodos

//Declaração

var texto = *"Algum valor dentro das aspas";*

//Retorna o tamanho da string

var tam = nome_string.length;

//Procura um termo na string e retorna a sua posição (índice)

//Se não encontrar, retorna -1

var indice = nome_string.indexOf(termo_para_presquisar);

//Procura um termo na string e retorna a sua posição (índice)

//Se não encontrar, retorna -1 | Obs.: a busca é realizada de trás pra frente

var indice = nome_string.lastIndexOf(termo_para_presquisar);

Strings

- Outros Métodos
 - SubString e Substituição

*//Retorna uma nova string considerando parte da string original
//e os parâmetros informados*

var outra_string = nome_string.substr(posicao_inicial, qtd_caracteres);

//Retorna uma nova string substituindo um termo por outro

var outra_string = nome_string.replace(termo_busca, termo_substituto);

//Retorna uma nova string substituindo um termo por outro (todos)

var outra_string = nome_string.replaceAll(termo_busca, termo_substituto);

Strings

- Outros Métodos
 - Maiúsculas e minúsculas

//Retorna uma nova string com todos os caracteres maiúsculos
var outra_string = nome_string.toUpperCase();

//Retorna uma nova string com todos os caracteres minúsculos
var outra_string = nome_string.toLowerCase();

//Acesso à apenas um caractere da string
var outra_string = nome_string[0]; *//Retorna o primeiro caractere*
var outra_string = nome_string[1]; *//Retorna o segundo caractere*
console.log(nome_string[5]); //Imprime o sexto caractere
console.log(nome_string[nome_string.length - 1]); //Imprime o último caractere

Concatenação de strings

- É utilizada para concatenar strings a outros dados

```
var x = "texto";  
var y = "palavra";  
var z = 5;
```

```
var a = x + y; //Resultado: textopalavra  
var b = x + y; //Resultado: texto palavra  
var c = x + z; //Resultado: texto5
```

Converter string para número

- É necessário para fazer cálculos entre números

```
var x = "1" + 2;  
//Resultado: 12
```

```
var y = parseInt("1") + 2;  
//Resultado: 3
```

```
/* parseFloat converte para o tipo float
```