

```
1 //MÓDULO: JS - STRINGS
2 //Capítulo: 02- (04 e 05) - STRINGS
3
4 console.log("Um string literal pode ficar entre aspas duplas");
5 console.log('Um string literal pode ficar entre aspas simples');
6 console.log(`Um string literal pode ficar entre crases`);
7
8 //`ENTRE CRASE CAMA-SE INTERPOLAÇÃO`
9
10 console.log("Concatenação e interpolação");
11 const num = 14.5;
12 console.log("O valor do produto é " + num);
13 console.log('O valor do produto é ' + num);
14 console.log(`O valor do produto é ${num}`);
15
16 console.log("Conversão entre number e string");
17 //PARA CONVERTER UM TIPO "number" PARA "string" - FUNÇÃO ".toString()"
18 const str1 = num.toString();
19 console.log(`Tipo de ${str1}: ${typeof str1}`);
20
21 //`${ inserir aqui uma expressão: pode ser uma variável por exemplo}
22 //No exemplo abaixo dentro da ${} podemos ver dois valores: Uma variável "str2" e uma
23 // expressão
24 // "typeof"
25 // ".toFixed(2) -> Para converter o num em string com duas casas decimais."
26 const str2 = num.toFixed(2);
27 console.log(`Tipo de ${str2}: ${typeof str2}`);
28
29 // Para converter uma String em Number:
30 const num1 = Number("14.5");
31 console.log(`Tipo de ${num1}: ${typeof num1}`);
32
33 //".parseInt() -> Também utilizado para converter uma String em Number:"
34 //Observação: o ",10" (Segundo argumento dentro da Função parseInt), INDICA A BASE DA
35 // CONVERSÃO
36 //DESTA VARIÁVEL... O 10 - Significa: Base 10 ou DECIMAL
37 // NESTE EXEMPLO O RETORNO SERÁ O PRÓPRIO: 21
38 const num2 = parseInt("21", 10);
39 console.log(`Tipo de ${num2}: ${typeof num2}`);
40
41 //NESTE CASO O ",16" - Significa: Base 16 ou HEXADECIMAL
42 //NESTE EXEMPLO O RETORNO SERÁ: 33 que é o 21 na base Hexadecimal
43 const num3 = parseInt("21", 16);
44 console.log(`Tipo de ${num3}: ${typeof num3}`);
45
46 //PARA CONVERTER UM NÚMERO FRACIONÁRIO (Com ponto flutuante em STRING)
47 //O "parseFloat" também utiliza o segundo argumento para a indicação da Base:
48 //DECIMAL, HEXADECIMAL etc.
49 const num4 = parseFloat("21.34", 10);
50 console.log(`Tipo de ${num4}: ${typeof num4}`);
51
52 console.log("Funções de string");
53
54 const str = "Maria Silva ";
55 console.log(`const str = "Maria Silva "`);
56 console.log(`toLowerCase: ${str.toLowerCase()}`);
57 console.log(`toUpperCase: ${str.toUpperCase()}`);
58 console.log(`charAt: ${str.charAt(3)}`); //Ele também aceita "${str[3]}"
```

```
57
58 // "REPLACE" - Neste exemplo irá substituir o "i" pelo "$"
59 // OBSERVAÇÃO: Desta forma o Replace irá substituir apenas a primeira ocorrência;
60 console.log(`replace (primeiro): ${str.replace("i", "$", )}`);
61 // DESTA FORMA, UTILIZANDO O "/i/g" O REPLACE IRÁ SUBSTITUIR TODAS AS OCORRÊNCIAS DO
  "i" POR "$".
62 console.log(`replace (todos): ${str.replace(/i/g, "$", )}`);
63
64 //Propriedade "LENGTH", UTILIZADO PARA FAZER A CONTAGEM DOS CARACTÉRIES DA STRING
65 //OBSERVAÇÃO: Os espaços também são incluídos na contagem.
66 console.log(`length: ${str.length}`);
67
68 // "indexOf" : Indica a PRIMEIRA ocorrência do que estiver indicado no argumento no
  exemplo a letra "i"
69 console.log(`indexOf: ${str.indexOf("i")}`);
70
71 // "lastIndexOf" : Indica a ÚLTIMA ocorrência do que estiver indicado no argumento no
  exemplo a letra "i"
72 console.log(`lastIndexOf: ${str.lastIndexOf("i")}`);
73
74 // UTILIZADO PARA RECORTAR UM PEDAÇO DA STRING
75 //Esta função possui dois argumentos (arg1, arg2)
76 //O "arg1" É a posição inicial
77 //O "arg2" OPCIONALMENTE É A POSIÇÃO FINAL
78 //Porque tem as posições: inicial e final?
79 //Por que se utilizar apenas a posição inicial, a String será cortada desta posição
  em diante.
80 //Utilizando a posição final, COMO NO EXEMPLO "8", esta função se limitaria em
  iniciar o corte
81 // na posição 3 e finalizar na posição 8. Dispensando todo o excesso anterior e
  posterior ao (3 e ao 8)
82 //Observação, no caso do 8 ele irá exibir até o caracter da posição 7... Da oitava em
  diante será
83 //suprimido.
84 console.log(`substring: ${str.substring(3, 8)}`);
85
86 console.log(`Exemplo Prático de Utilização da Função Substring`);
87 const data = "14/10/2021";
88 const mes = data.substring(3, 5);
89 console.log(mes)
90
91 // OUTRO EXEMPLO: Suponha que vc tenha um valor de um preço no formato internacional.
92 //Por exemplo: "200.99".
93 //Suponha que eu queira utiliza-lo com a vírgula e não com o ponto.
94 //Neste caso podemos utilizar o REPLACE.
95 //Veja:
96
97 //Observe que aqui o valor 200.99 está em forma de número "number"
98 const valor = 200.99
99
100 //Agora iremos converter este número para String com o ".toString()"
101 //Em seguida iremos substituir o "." pela "," com o ".replace(".", ",")
102 const novoValor = valor.toString().replace(".", ",");
103 console.log(novoValor);
104
105
106
107
108
109
```

```
110 //OUTRO EXEMPLO: Utilizar o ".toLowerCase()" no e-mail de alguém.  
111 //Suponha que alguém tenha cadastrado o e-mail em um formulário com letras  
    maiúsculas.  
112 //Certamente, você não vai querer armazená-lo assim.  
113 //Então, para este fim será necessário utilizar a função ".toLowerCase" para  
    converter o string  
114 //do e-mail da pessoa em um string com todas as letras minúsculas.  
115  
116 //TRIM - A função ".trim()", tem como objetivo APAGAR OS ESPAÇOS EM BRANCO DA STRING.  
    NO COMEÇO E NO FIM.  
117 console.log(`trim: ${str.trim()}`);
```