



Capítulo 6

Strings e Datas

Programação com JavaScript & TypeScript

Ambiente para praticar

Use o TypeScript Playground para testar código direto no navegador

<https://www.typescriptlang.org/play/>

1. Escreva código TypeScript no painel esquerdo
2. Use `console.log()` para ver saída na aba "Logs"
3. Clique em "Run" para executar
4. Também funciona com JavaScript puro!

Acessar caracteres e tamanho

Cada caractere de uma string tem um índice (começa em 0)

J	a	v	a	S	c	r	i	p	t
0	1	2	3	4	5	6	7	8	9

charAt(posição)

```
const s = "JavaScript";  
  
s.charAt(0) // "J"  
s.charAt(4) // "S"  
s[0]       // "J"  
// Atalho moderno com [ ]
```

.length

```
const s = "JavaScript";  
  
s.length // 10  
  
// Último caractere:  
s.charAt(s.length - 1) // t
```

Espaço para exemplo ao vivo

Converter maiúsculas e minúsculas

Útil para comparar textos sem diferenciar caixa

toUpperCase()

Converte tudo para MAIÚSCULAS

```
"café".toUpperCase()  
// "CAFÉ"
```

toLowerCase()

Converte tudo para minúsculas

```
"CAFÉ".toLowerCase()  
// "café"
```

Uso prático – comparar sem diferenciar caixa

```
const resposta = "Sim";  
  
// Errado: "Sim" !== "sim"  
if (resposta === "sim") { ... }  
  
// Correto: converte antes de comparar  
if (resposta.toLowerCase() === "sim") { ... }
```

Extrair partes de uma string

`substring(início, fim)` — extrai do início até fim (não incluso)

```
const cpf = "012.345.678-90";  
  
// Extrair partes do CPF  
cpf.substring(0, 3)  
// "012"  
  
cpf.substring(4, 7)  
// "345"  
  
// Do índice 4 até o final  
cpf.substring(4)  
// "345.678-90"
```

Dicas

1

O 1º parâmetro é o índice inicial (incluído)

2

O 2º parâmetro é o índice final (não incluído)

?

Se omitir o 2º, vai até o final da string

≡

Strings são imutáveis: retorna uma nova string

Espaço para exemplo ao vivo

Localizar conteúdo em strings

`indexOf()`

Posição da 1ª ocorrência (ou -1)

```
const s = "abcabc";  
s.indexOf("b") // 1  
s.indexOf("z") // -1
```

`lastIndexOf()`

Posição da última ocorrência (ou -1)

```
const s = "abcabc";  
s.lastIndexOf("b") // 4  
s.lastIndexOf("z") // -1
```

`includes()` – alternativa moderna

```
const email = "aluno@cefet.edu.br";  
  
email.includes("@") // true  
email.includes("gmail") // false
```

Dividir string em vetor

split(separador) transforma uma string em um array

Dividir por espaço

```
const frase = "Olá Mundo JS";  
  
const arr = frase.split(" ");  
  
// ["Olá", "Mundo", "JS"]  
// arr.length → 3
```

Dividir por delimitador

```
const data = "18/05/2026";  
  
const p = data.split("/");  
  
// ["18", "05", "2026"]  
// p[0]→"18" p[1]→"05"
```

Dividir cada caractere

```
"ABC".split("") // ["A", "B", "C"] – separador vazio = cada caractere
```

Espaço para exemplo ao vivo

Validar e substituir texto

match() para validar padrões e replace() para substituir

match() – validar senha

```
const senha = "Abc@123";

// Tem letra maiúscula?
senha.match(/[A-Z]/)

// Tem número?
senha.match(/[0-9]/)

// Tem caractere especial?
senha.match(/[!@#%]/)

// Retorna null se não encontrar
```

replace() – substituir texto

```
// Substituir primeira ocorrência
"abcabc".replace("a","X")
// "Xbcabc"

// Substituir TODAS com regex /g
"abcabc".replace(/a/g,"X")
// "XbcXbc"

// Ou use replaceAll()
"abcabc".replaceAll("a","X")
// "XbcXbc"
```

Espaço para exemplo ao vivo — Validação de senha

Manipulação de datas

O objeto Date permite criar e manipular datas e horários

Criar datas

```
// Data e hora atuais
const agora = new Date();

// Data específica (ano,mês-1,dia)
const natal =
  new Date(2026, 11, 25);
// Mês: 0=Jan ... 11=Dez
```

Extrair informações

```
const d = new Date();

d.getFullYear() // 2026
d.getMonth()    // 0-11
d.getDate()     // 1-31
d.getHours()    // 0-23
d.getMinutes()  // 0-59
```

Formatar e exibir

```
const d = new Date();
d.toLocaleDateString("pt-BR") // "18/05/2026"
d.toLocaleTimeString("pt-BR") // "14:30:00"
d.toLocaleString("pt-BR")    // "18/05/2026, 14:30:00"
```

Resumo – Capítulo 6

<code>charAt / []</code>	Acessar caractere pela posição
<code>.length</code>	Número de caracteres da string
<code>toUpperCase</code>	Converte para MAIÚSCULAS
<code>toLowerCase</code>	Converte para minúsculas
<code>substring()</code>	Extrai parte da string (início, fim)
<code>indexOf()</code>	Posição da 1ª ocorrência ou -1
<code>split()</code>	Divide string em array por separador
<code>match()</code>	Valida padrões com expressão regular
<code>replace()</code>	Substitui texto na string
<code>new Date()</code>	Cria e manipula datas e horários