

# Strings METHODS

length

charAt

0	1	2	3	4	5	6
A	I	b	e	+	+	o

N = 7

→ `charAt(2) = "b"`

→ index of "+" = 4 involve all letters

se non trova nulla  
ritorna -1

→ includes(...)

```
nome.includes("betr")
```

```
false
```

```
nome.includes("bert")
```

```
true
```

È come se  
utilizzare  
indexOf(...) se da  
ritorna un valore  
booleano (true, false)

→ lastIndexOf(...)

```
"elena"
```

```
nome.indexOf("e")
```

```
0
```

```
nome.lastIndexOf("e")
```

```
2
```

Cerca  
l'ultima  
corrispondenza

→ replace (Lettera da cercare, Lettera da scrivere)

```
nome.replace("a", "z")
```

```
"elenz"
```

```
nome.replace("e", "z")
```

```
"zlenu"
```

impresza sb  
le 1<sup>a</sup>  
corrispondenza

```
nome.replace("a", "z")
```

```
"elenz"
```

```
nome
```

```
"elena"
```

HA 2  
PARAMETRI

"elenz"  
↑  
nome = nome.replace("a", "z")  
└─┬─┘  
"elenu"

→ substr(par1, par2) ritorna un pezzo  
di stringa

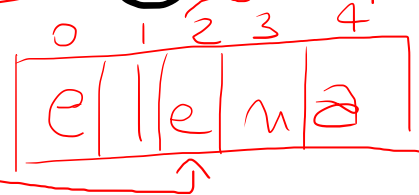
"elena"

nome.substr(2, 3)

"ena"

nome.substr(1, 2)

"le"



1° PAR  
POSIZIONE  
DI PARTENZA

2° PAR  
N° DI LETTERE  
DA ESTRARRE



trim()

"e l e n a"

nome = " elena "

" elena "

nome.trim()

"elena"

Elimina spazi vuoti  
da inizio e fine  
della stringa

→ split(pat+1) → split(" ")

var testo = "Quel ramo del lago";  
↓ ↓ ↓ ↓  
["Quel", "ramo", "del", "lago"]  
0 1 2 3

Converte una  
stringa in array  
dividendo la  
parte in base  
al testo  
passato come  
parametro

var testo = "A;B;C;D"  
testo.split(";") → ["A", "B", "C", "D"]  
0 1 2 3