

```
//OPERADORES UNÁRIOS:
//Operadores unários são operadores que operam sobre 1 único operando. Entre os operadores unários temos 2 tipos: Prefix e Posfix, onde os unários Prefix têm precedência sobre os Posfix.

//EXEMPLOS DE OPERADORES UNÁRIOS:
let n1 = 1;
let n2 = 2;
//Usando o operador de autoincremento ++:
console.log('1', n1++); //Perceba que, quando usamos o Posfix o número não incrementa no momento da declaração...
console.log('1', n1); //O incremento só acontece depois...
console.log('1', ++n1); //Quando usamos o Prefix, o incremento acontece na hora...

//usando o operador de autodecremento --:
console.log('\n2', n2--); //Perceba que, quando usamos o Posfix o número não decrementa no momento da declaração...
console.log('2', n2); //O decremento acontece só depois...
console.log('2', --n2); //Quando usamos Prefix o decremento acontece na hora...

//Operadores de negação e positividade lógica também são unários:
console.log('\n3', !n2); //O n2 neste momento está valendo "0" por causa dos decrementos acima, como isso quando invertemos o valor lógico o resultado é "true"...
console.log('3', !!n2); //Quando usamos o verificador !! ele mostra que o valor lógico real de n2 é false por ser o valor "0"...

//Operador typeof:
console.log('\n4', typeof n1); //O typeof também é um operador usado para conhecer o tipo de um elemento...
console.log('4', typeof n2);

//Operador delete:
//O operador delete é usado para apagar valores dentro de objetos e arrays...
let objeto = {frase: 'Oi eu existo', frase2: 'Até agora'};
console.log('\n5', objeto);

delete objeto.frase2; //Usando delete para apagar uma chave dentro de um objeto...
console.log('5', objeto);
delete objeto.frase; //Usando delete para apagar outra chave de um objeto...
console.log('5', objeto);
```

RESULTADO NO CONSOLE...

```
[Running] node "c:\Users\User\Documents\JAVASCRIPT\ARQUIVOS_DAS_AULAS\049-Operadores_Unarios.js"
```

```
1) 1
```

```
1) 2
```

```
1) 3
```

```
2) 2
```

```
2) 1
```

```
2) 0
```

```
3) true
```

```
3) false
```

```
4) number
```

```
4) number
```

```
5) { frase: 'Oi eu existo', frase2: 'Até agora' }
```

```
5) { frase: 'Oi eu existo' }
```

```
5) {}
```

```
[Done] exited with code=0 in 0.163 seconds
```