# Comparação de JavaScript e Operadores Lógicos

[❮ Anterior](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_booleans.asp&usg=ALkJrhgJ9yVhiJDIueZKx5YuQRHnZgBcMQ)[Próximo ❯](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_if_else.asp&usg=ALkJrhjSgh__jTJN29HqGnBuJ9DLi4eM1g)

Os operadores de comparação e logs são usados ​​para testar true ou false .

## Operadores de comparação

Os operadores de comparação são usados ​​em declarações lógicas para determinar igualdade ou diferença entre variáveis ​​ou valores.

Dado que **x = 5** , a tabela abaixo explica os operadores de comparação:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Operator** | **Description** | **Comparing** | **Returns** | **Try it** |
| == | equal to | x == 8 | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison1&usg=ALkJrhhAQ9063tym0FVZZ0sdvxImaGgMlQ) |
| x == 5 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison2&usg=ALkJrhgJ3Qt9PMZoUnwD4vlMQFVx3WTPpw) |
| x == "5" | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison12&usg=ALkJrhjd50QbwLgCsn0NME18KQFtMy2Ghg) |
| === | equal value and equal type | x === 5 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison4&usg=ALkJrhjeklPpyKZYatUK-pYzPy2tSf7Amw) |
| x === "5" | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison3&usg=ALkJrhiN9bzBhs-P7rrnR7ZiQNsLHWG3PA) |
| != | not equal | x != 8 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison5&usg=ALkJrhhi1LMYBltX-_Sw4uBQicGd1kIDzw) |
| !== | not equal value or not equal type | x !== 5 | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison7&usg=ALkJrhhdK1Rfg5wiNylt4dm8G1MtU2Lj8g) |
| x !== "5" | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison6&usg=ALkJrhhEe0LOLtS2N29lsMEiOh5ffVlV8A) |
| x !== 8 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison13&usg=ALkJrhgosTcij1xOd5MCOH_dlkrfQXLFsg) |
| > | greater than | x > 8 | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison8&usg=ALkJrhjStLm7AIAbFpmiHTRBfJcWW6Ko9w) |
| < | less than | x < 8 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison9&usg=ALkJrhg824wtUKsoLt6cuy3jmM9MwV9C2w) |
| >= | greater than or equal to | x >= 8 | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison10&usg=ALkJrhiloKQbsT_L9mHOrtbleBULA5zHmw) |
| <= | less than or equal to | x <= 8 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison11&usg=ALkJrhjBmZUVeZcAfuwp1X82g44Zx8meGQ) |

## Como pode ser usado

Os operadores de comparação podem ser usados ​​em declarações condicionais para comparar valores e agir de acordo com o resultado:

if (age < 18) text = "Too young";

Você aprenderá mais sobre o uso de declarações condicionais no próximo capítulo deste tutorial.

## Operadores lógicos

Os operadores lógicos são usados ​​para determinar a lógica entre variáveis ​​ou valores.

Dado que **x = 6** e **y = 3** , a tabela abaixo explica os operadores lógicos:

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Description** | **Example** | **Try it** |
| && | and | (x < 10 && y > 1) is true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_and&usg=ALkJrhiPavnwLh_72uyolaW06w2D8B4NLQ) |
| || | or | (x == 5 || y == 5) is false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_or&usg=ALkJrhjoIPM6Qe2CiFkTt1azhYD-re0Egg) |
| ! | not | !(x == y) is true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_not&usg=ALkJrhiqCDt2VP06dOZGsh9jfqRrRB8PdA) |

## Operador Condicional (Ternário)

O JavaScript também contém um operador condicional que atribui um valor a uma variável com base em alguma condição.

### Sintaxe

variablename = ( condition ) ? value1 : value2

### Exemplo

var voteable = (age < 18) ? "Too young":"Old enough";

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison&usg=ALkJrhh8ayNPOX7ngyWGBHBWeCRUlfgA9A)

Se a idade da variável for inferior a 18, o valor da variável votável será "Demasiado jovem", caso contrário, o valor de votar será "antigo o suficiente".

## Comparando diferentes tipos

A comparação de dados de diferentes tipos pode dar resultados inesperados.

Ao comparar uma string com um número, o JavaScript irá converter a string em um número ao fazer a comparação.Uma seqüência vazia converte para 0. Uma seqüência não-numérica converte para NaN, o que é sempre falso.

|  |  |  |
| --- | --- | --- |
| **Case** | **Value** | **Try** |
| 2 < 12 | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_20&usg=ALkJrhgyLzZw4bU89OUo56HrjyVA3C48bA) |
| 2 < "12" | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_21&usg=ALkJrhjNNecsh6cYc13GFxv88XWUVTRNqA) |
| 2 < "John" | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_23&usg=ALkJrhjiif9zz-4N1_-RiOKKKR7CdB2BlA) |
| 2 > "John" | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_24&usg=ALkJrhg-kwMc_-LzGmTsoxTG5v1KiMaO8w) |
| 2 == "John" | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_25&usg=ALkJrhgNFjsPumFtkb2kbP3itnqTmAGrew) |
| "2" < "12" | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_26&usg=ALkJrhjBuoADl0pifcIVFkWSGX_KPx2AtQ) |
| "2" > "12" | true | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_27&usg=ALkJrhh2tmkLBytxzoG6_pZuO47oVhKkgg) |
| "2" == "12" | false | [Try it »](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_comparison_28&usg=ALkJrhhHH9e7X7mNFO2rGYXiAbsbr9D6uA) |

Ao comparar duas cordas, "2" será maior que "12", porque (em ordem alfabética) 1 é inferior a 2.

Para garantir um resultado adequado, as variáveis ​​devem ser convertidas para o tipo adequado antes da comparação:

age = Number(age);  
if (isNaN(age)) {  
    voteable = "Input is not a number";  
} else {  
    voteable = (age < 18) ? "Too young" : "Old enough";  
}