



IN

INFINITY SCHOOL

V I S U A L A R T C R E A T I V E C E N T E R

JS – Condicionais em JavaScript

01

Operadores

- Operadores lógicos
- Operadores de comparação

02

Condicionais

03

Exercícios




JS – Operadores Lógicos

São usados para realizar operações lógicas. Elas podem ser do tipo AND, OR e NOT. Os operandos devem ser lógicos, verdadeiro ou falso. Também podem operar sobre expressões lógicas, ou seja, que retornem valores verdadeiro ou falso.

– Operador AND (&&)

O operador AND (&&) recebe dois operandos e retorna verdadeiro se, e somente se ambos os operandos sejam verdadeiros. Retorna falso, caso contrário.



```
1 <script>
2     let a=3
3     let b=5
4     a==3 && b<10
5         //retorna verdadeiro ou true
6
7 </script>
```


JS – Operadores Lógicos

– Operador OR (||)


Para o operador OR (||) retornar verdadeiro, basta que um dos operandos seja verdadeiro. Ele também retorna verdadeiro caso os dois operandos sejam verdadeiros. Retorna falso, se os dois forem falsos.

– Operador NOT (!)

O operador de negação (!) é um operando unário, isto é, opera sobre apenas um operando. Ele nega, inverte o valor lógico do operando.



```
1 <script>
2   let a=3
3   let b=5
4   a==3 || b<10
5       //retorna verdadeiro ou true
6
7 </script>
```



```
1 <script>
2   let a=3
3   let b=5
4   !(a==3)
5       //retorna falso ou false
6
7 </script>
```


JS – Operadores de Comparação

- Imagine a seguinte situação...

Você tem dois valores numéricos e quer comparar ambos para ver qual deles é maior. Para isso, o JavaScript nos traz os **operadores de comparação**, eles são meios que a linguagem nos traz para comparar valores. São eles:

==: Igualdade abstrata – Verifica se um valor é igual ao outro.

===: Igualdade estrita – Verifica se o valor e tipo são iguais.

!=: Diferença abstrata – Verifica se um valor é diferente de outro.

!==: Diferença estrita – Verifica se o valor ou tipo são diferentes.

>: Maior que – Verifica se um valor é maior que outro.

<: Menor que – Verifica se um valor é menor que outro.

>=: Maior igual – Verifica se um valor é maior ou igual ao outro.

<=: Menor igual – Verifica se um valor é menor ou igual ao outro.



JS – Operadores de Comparação



```
1 <script>
2 //Declaração de variáveis
3 let numero1=10//variável tipo inteira
4 let valor1= '10'//variável tipo string
5
6 //saída será true pois numero1 e valor1 possuem o mesmo valor
7 console.log(numero1==valor1)
8 //saída será false pois numero1 e valor1 possuem o mesmo valor mas não o mesmo tipo
9 console.log(numero1===valor1)
10 //saída será false pois numero1 e valor1 possuem o mesmo valor
11 console.log(numero1!=valor1)
12 //saída será true pois numero1 e valor1 possuem o mesmo valor mas não o mesmo tipo
13 console.log(numero1!==valor1)
14 //saída será false pois numero1 e valor1 possuem o mesmo valor
15 console.log(numero1>valor1)
16 //saída será false pois numero1 e valor1 possuem o mesmo valor
17 console.log(numero1<valor1)
18 //saída será true pois numero1 e valor1 possuem o mesmo valor
19 console.log(numero1>=valor1)
20 //saída será true pois numero1 e valor1 possuem o mesmo valor
21 console.log(numero1<=valor1)
22 </script>
```


JS – Operadores de Comparação

Esses operadores sempre vão nos retornar um valor booleano, ou seja, verdadeiro ou falso, a depender do resultado da comparação. Como no exemplo:

```
1 // Define o número 1 como 5
2 let numero1 = 5
3 // Pede ao usuário para que digite o valor de número 2
4 let numero2 = prompt('Digite um número: ')
5
6 // Se o numero 1 for maior ou igual ao valor digitado pelo usuário,
7 // o valor de resultado vai ser igual a TRUE, caso contrário será igual a FALS
8 resultado = numero1 >= numero2
9 // Mostra em tela o valor de resultado
10 console.log(resultado)
```

JS – Condicionais

As estruturas condicionais estão ligadas à tomada de decisão de um algoritmo. Ao utilizar expressões que retornam verdadeiro ou falso, o algoritmo executa o bloco de comandos relativos a este resultado.

Vamos aprender a utilizar as estruturas condicionais no JavaScript, começando pelo uso do if(se), else if e else(senão).

```
1  <script>
2  if (condição1){
3      instrução1
4  }
5  else if (condição2){
6      instrução2
7  }
8  else if (condição3){
9      instrução3
10 }
11 ...
12 else{
13     instruçãoN
14 }
15
16 </script>
```


JS – Condicionais

if: A estrutura if é a forma mais básica de uma estrutura condicional. Ela avalia uma condição e executa um bloco de código se a condição for verdadeira (ou seja, seu valor é true).

else if: A estrutura else if é usada quando você quer verificar mais de uma condição, além da condição inicial do if. Se a condição do if não for verdadeira, o código verificará a condição do else if subsequente e, se essa condição for verdadeira, executará o bloco de código associado.

else: A estrutura else é usada junto com o if e/ou else if. Se todas as condições anteriores forem falsas, o bloco de código associado ao else será executado.

JS – Condicionais

```
1 <script>
2     /*utilizamos a palavra if seguido da condição idade > 18 e
3     else caso a condição do if não seja satisfeita.*/
4     let idade = 19
5     if(idade>18){
6         /* caso a variável idade seja maior que 18 (verdadeiro),
7         os comandos que estarão no bloco serão executados*/
8         console.log('Você é obrigado a votar')
9     }else{
10         // senão o comando do else será satisfeita
11         console.log('Você não é obrigado a votar')
12     }
13 </script>
```

JS – Condicionais

Ternários Condicionais

Os ternários condicionais facilitam o código, tornando o código mais curto, frequentemente usada como um atalho para a instrução if.

Uma diferença crucial em relação ao if é que o ternário é um operador e não uma declaração tipo função ou variável. Ele sempre resulta em um valor que é retornado conforme sua lógica.

A vantagem do ternário para atribuição de valor é clara: não é preciso repetir o nome da variável. Por exemplo, nesta atribuição com if usamos 2 linhas (incluindo a declaração da variável):

```
<script>  
  let idade=19  
  idade>18?console.log('Você é obrigado a votar'):console.log('Você não é obrigado a votar')  
</script>
```


JS – Condicionais

Ternários Condicionais

condicao ? valorSeVerdadeiro : valorSeFalso;

condicao: Uma expressão que será avaliada. Se o resultado for verdadeiro (ou seja, seu valor é true), o valorSeVerdadeiro será retornado; caso contrário, o valorSeFalso será retornado.

valorSeVerdadeiro: O valor retornado caso a condição seja verdadeira.

valorSeFalso: O valor retornado caso a condição seja falsa.



```
1 <script>
2   let idade = 20;
3   let mensagem = idade >= 18 ? "Você é maior de idade." : "Você é menor de idade.";
4   console.log(mensagem); // Saída: "Você é maior de idade."
   </script>
```

JS – Condicionais

Switch

O switch é uma outra estrutura condicional em JavaScript que permite avaliar o valor de uma expressão e executar um bloco de código correspondente ao valor dessa expressão. É especialmente útil quando você tem várias condições para verificar e deseja evitar várias declarações if, else if e else

JS – Condicionais

Switch



```
1  <script>
2      let nome= 'raama'
3
4      switch(nome){//A expressão cujo valor será comparado com os casos.
5          case 'joao':
6              /*Case: O uso de condicionais permite que um bloco de código seja
7               executado se a expressão for igual a um valor específico.*/
8              console.log("o nome é João")
9              break;
10             /*O comando "break" é utilizado para sair do bloco switch após
11              a execução do código correspondente a um caso.*/
12          case 'raama':
13              console.log("o nome é Raama")
14              break;
15          case 'josé':
16              console.log("o nome é José")
17              break;
18          default:
19              //É opcional e representa o caso em que nenhum dos valores dos casos corresponde à expressão.
20              console.log("Nenhum dos nomes")
21      }
22
23  </script>
```


JS – Condicionais

Switch



```
1  <script>
2      let escolha=3
3      switch(escolha){
4          case 0:
5              console.log('Valor 0');
6              break;
7          case 1:
8              console.log('Valor 1');
9              break;
10         case 2:
11             console.log('Valor 2');
12             break;
13         case 3:
14             console.log('Valor 3');
15             break;
16         case 4:
17             console.log('Valor 4');
18             break;
19     }
20 </script>
```

JS – Exercícios

- Crie uma aplicação capaz de identificar a faixa etária com base na idade informada pelo usuário. Considere os seguintes critérios: Se a idade informada for maior ou igual a 0 e menor que 15, exibir a mensagem “Criança”. Se a idade informada for maior ou igual a 15 e menor que 30, exibir a mensagem “Jovem”. Se a idade informada for maior ou igual a 30 e menor que 60, exibir a mensagem “Adulto”. Se a idade informada for maior ou igual a 60, exibir a mensagem “Idoso”
- Você é o responsável por um sistema de login em um site. Para fazer login, o usuário precisa inserir o nome de usuário e a senha corretos. Defina uma senha e email padrão e solicite que o usuário insira o email e senha e verifique se está igual ao email e senha definido no escopo do código. Se as senhas e os emails coincidirem mostre no console “Login efetuado com sucesso”, senão mostre no console “Senha ou email incorretos”.
OBS: UTILIZE O PROMPT PARA PEDIR A SENHA E O EMAIL PARA O USUÁRIO

JS – Exercícios

- Você é um desenvolvedor de um aplicativo de controle de acesso e você precisa verificar o acesso. Peça para o usuário para digitar qual o nível de acesso do usuário dele: (sendo 1 para "visitante", 2 para "membro" e 3 para "administrador"). Implemente o programa utilizando o switch para verificar o nível de acesso e retornar a seguinte mensagem de acordo com o valor do parâmetro:
 - Caso o nível de acesso seja 1: mostre no console "Acesso de Visitante. Recursos limitados."
 - Caso o nível de acesso seja 2: mostre no console "Acesso de Membro. Recursos básicos."
 - Caso o nível de acesso seja 3: mostre no console "Acesso de Administrador. Todos os recursos."
 - Se o nível de acesso não for nenhum dos valores válidos (1, 2 ou 3), mostre no console: "Nível de acesso inválido. Verifique o valor inserido."

OBS: UTILIZE O PROMPT PARA OBTER A INFORMAÇÃO DO NÍVEL DE ACESSO INFORMADA PELO USUÁRIO



IN

INFINITY SCHOOL

V I S U A L A R T C R E A T I V E C E N T E R

IN PARABÉNS! VOCÊ TERMINOU A AULA 02 DO MÓDULO DE JAVASCRIPT