

EXERCÍCIO DIRIGIDO

Estruturas de Decisão

Em TypeScript, assim como em qualquer linguagem de programação, na maioria das vezes é necessário tomar decisões. Um computador padrão só reconhece dois estados lógicos: False (false) ou Verdadeiro (true). Estes são os resultados possíveis em toda e qualquer “decisão” que um script tiver que executar.

Vale aqui relembrar os operadores relacionais e lógicos disponíveis:

Operadores Relacionais

==	Igual a
<	Menor que
>	Maior que
<=	Menor ou igual a
>=	Maior ou igual a
!=	Não igual (diferente)

São utilizados em expressões lógicas para se testar a relação entre dois valores do mesmo tipo.

Exemplos: `3 == 3` (3 é igual a 3?) resulta em **true**; `"A" > "B"` ("A" está antes de "B" na ordem alfabética?) resulta em **false**.

Importante: No TypeScript, as comparações entre *strings* **diferenciam** as letras maiúsculas das minúsculas. Assim, "ABC" não é igual a "abc". Valores lógicos obedecem à seguinte ordem: false (false) < true (verdadeiro).

Operadores Lógicos

!	Operador unário de negação: ! true = false , e ! false = true . Tem a maior precedência entre os operadores lógicos.
	Operador or (ou) que resulta true quando um dos seus operandos lógicos for true.
&&	Operador and (e) que resulta true somente se seus dois operandos lógicos forem true.

Vamos relembrar a Tabela Verdade:

A	B	! A	! B	A B	A && B
true	true	false	false	true	true
true	false	false	true	true	false
false	true	true	false	true	false
false	false	true	true	false	false

Vejamos alguns exemplos:

Para A = 5, B = 3, C = 1, D = "A" e F = "B"

A > B = true	A > B C > A = true	A < B && C > A = false
A < B = false	A < B C > A = false	! (A > C) = false
A >= C = true	A > B && C > A = false	A > B B > C => true
D == F = false	D != F = true	D < F = true

Estrutura condicional if / else

A sintaxe é a seguinte:

```
if (<expressão-lógica>) {  
    <sequência-de-comandos>  
}
```

Ao encontrar este comando, o TypeScript analisa a <expressão-lógica>. Se o resultado for **true**, todas as linhas da <sequência-de-comandos> (entre esta linha e a linha com **}**) serão executadas. Se o resultado for **false**, estas linhas são desprezadas e a execução do script continua a partir da primeira linha após a chave de fechamento do bloco.

```
if (<expressão-lógica>) {  
    <sequência-de comandos-1>  
} else {  
    <sequência-de-comandos-2>  
}
```

Nesta outra forma de estrutura, se o resultado da avaliação da <expressão-lógica> for **true**, todas as linhas da <sequência-de-comandos-1> (entre esta linha e a linha com **else**) são executadas, e a execução continua a partir da primeira linha depois da última chave de fechamento. Se o resultado for **false**, estas linhas são desprezadas e o script continua a ser executado a partir da primeira linha depois do **else**, executando todas as linhas da <sequência-de-comandos-2> (até a linha com a chave de fechamento).

- 1) O Script abaixo compara o valor de duas variáveis inteiras e de acordo com o resultado mostrará em no console padrão uma mensagem ou outra:

```
1 let A:number, B:number;  
2  
3 A = 5;  
4 B = 3;  
5  
6 if (A > B) {  
7     console.log("A comparação é verdadeira!");  
8 } else {  
9     console.log("A comparação é falsa!");  
10 }
```

Copie e cole o script anterior em um arquivo .ts dentro do seu projeto e execute-o: Veja o resultado. Inverta os valores das variáveis A e B e execute-o novamente. Veja o resultado.

Perceba que se a comparação de $A > B$ for verdadeira, será mostrada na tela a primeira frase. Caso contrário, será mostrada a segunda frase.

Vamos alterar agora o exercício dado na lista anterior, para que, baseado em uma tabela, ele informe no console o status do IMC do usuário:

- 2) O Índice de Massa Corporal (IMC) é uma medida que indica se uma pessoa está abaixo, dentro ou acima do seu peso ideal, de acordo com a seguinte tabela:

IMC	Indicação
$\leq 18,5$	Abaixo do peso ideal.
$18,5 \leq 24,9$	Dentro do peso ideal.
$25,0 \leq 29,9$	Acima do peso ideal.
$30,0 \leq 34,9$	Obesidade Grau I
$35,0 \leq 39,9$	Obesidade Grau II
$\geq 40,0$	Obesidade Grau III

Para calculá-lo utiliza-se a seguinte fórmula: $\frac{\text{peso (Kg)}}{\text{altura}^2 \text{ (m)}}$

Faça um script que receba dois valores constantes, peso e altura, calcule e mostre no console o valor do IMC de uma pessoa. O script deverá ainda, baseado na tabela dada, informar ao usuário o status do seu peso ideal.

1	Crie um arquivo .ts para o script	exercicio01.ts
2	Crie 3 variáveis do tipo real para conter os valores constantes do peso e da altura de uma pessoa e o valor calculado do seu IMC.	<code>let peso:number, altura:number, imc:number;</code>
3	Atribua à variável peso o valor constante de 82 e à variável altura o valor de 1.63.	<code>peso = 82; altura = 1.63;</code>
4	Calcule o valor do IMC da pessoa seguindo a fórmula acima.	<code>imc = peso / Math.pow(altura, 2);</code>
5	Mostre em uma janela do navegador o resultado da operação.	<code>console.log("Seu IMC é: ",imc);</code>
6	Teste agora o valor do IMC de acordo com a tabela dada e envie a mensagem correspondente para o console.	<pre> if (imc <= 18.5) { console.log("Abaixo do peso ideal"); } else { if (imc <= 24.9) { console.log("Dentro do peso ideal"); } else { if (imc <= 29.9) { console.log("Acima do peso ideal"); } else { if (imc <= 34.9){ console.log("Obesidade grau I"); } else { if (imc <= 39.9){ console.log("Obesidade grau II"); } else { console.log("Obesidade grau III"); } } } } } </pre>
7	Salve e execute o script	<code>npm run ex01</code>

Vamos alterar agora, mais um exercício da lista anterior. O exercício calculava a média de três notas do aluno. Vamos inserir a necessidade de o script informar se o aluno ao atingir a média 6, estará aprovado. Caso contrário, será reprovado.

- 3) A média de um aluno em uma escola é calculada por três notas de provas que valem cada uma de 0 a 10, podendo conter valores decimais. Faça um script que receba os três valores constantes das notas de um aluno, calcule sua média simples e mostre no console além da média final, a informação que estará aprovado se sua média for superior ou igual a 6 e reprovado caso a média seja inferior a 6.

1	Crie um arquivo exercicio03.ts para o script
2	Crie 3 variáveis do tipo numérico para conter os valores constantes das notas e uma para o valor calculado da média delas.

3	Atribua às variáveis das notas os valores 7.5, 8, 9.
4	Calcule o valor da média do aluno.
5	Mostre no console o resultado da operação.
6	Verifique se a média do aluno é maior ou igual a 6. Se for, mostre no console a mensagem que está aprovado. Caso contrário, ou seja, se a média for inferior a 6, mostre a mensagem que está reprovado.
7	Salve e execute seu script

Entrada de dados no TypeScript

A entrada de dados via teclado (digitação) não é algo trivial no TypeScript, principalmente na programação *backend*, onde não se espera que tenha digitação de usuário. No entanto, existe algumas formas de fazê-lo, conforme mostrado no código a seguir:

```
var readline = require('readline');
var resp = "";

var leitor = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});

leitor.question("Digite alguma coisa?\n", function(answer:string) {
  var resp = answer;
  console.log("\nSua resposta '" + resp + "' foi grava com sucesso na variável resp");
  leitor.close();
});
```

Neste momento o script dará uma parada em sua execução e ficará aguardando que algo seja digitado pelo usuário no console, até que, após a digitação, se tecele ENTER. Assim, o valor digitado será capturado e armazenado na variável resp.

- 4) Faça um script que receba um valor digitado pelo usuário informando sua idade em anos. Depois, baseado na tabela a seguir informe no console em que categoria etária ele se encontra:

Idade (anos)	Categoria
<= 10	Criança
11 <= 13	Pré-Adolescente
14 <= 17	Adolescente
18 <= 59	Adulto
>= 60	Idoso

- 5) Escreva um script que solicite a um candidato a vereador que digite valores para o número total de eleitores de um município, o número de votos brancos, nulos e válidos, e quantos votos obteve na última eleição. Se o candidato tiver mais de 10% de votos da soma dos votos válidos mais votos brancos, informar no console que ele foi eleito. Informar também, qual o percentual que sua votação atingiu perante a quantidade de eleitores do município.
- 6) Escreva um script que solicite a um funcionário o valor do seu salário mensal atual. Calcule o salário atualizado desse funcionário, depois do reajuste, seguindo a regra: Se o salário for inferior ou igual a R\$ 650,00, o reajuste será de 10%. Caso o salário seja maior que R\$ 650,00, o reajuste será de apenas 5%. Mostre no console o salário original, o percentual de reajuste e o valor final do salário reajustado.
- 7) Dado três valores X, Y, Z, verificar se eles podem ser os comprimentos dos lados de um triângulo, e se forem, verificar se é um triângulo equilátero, isósceles ou escaleno. Se eles não formarem um triângulo, escrever uma

mensagem no console informando isso. Antes da elaboração do script, torna-se necessária a revisão de algumas propriedades e definições:

Propriedade – Para que seja possível formar um triângulo, o comprimento de cada lado deve ser menor do que a soma dos comprimentos dos outros dois lados.

Definição 1 - Chama-se triângulo equilátero os que tem os comprimentos dos três lados iguais;

Definição 2 - Chama-se triângulo isósceles o triângulo que tem os comprimentos de dois lados iguais;

Definição 3 - Chama-se triângulo escaleno ao triângulo que tem os comprimentos dos três lados diferentes.

Comando de Seleção Múltipla: switch/case

Existirão situações em que o comando **if** se mostrará complicado para a tomada de decisões dentro de um script, dependendo da quantidade de verificações necessárias a serem feitas.

Quando isso acontecer, existe um outro bloco de comandos que permitirá múltiplas seleções: A sintaxe é a seguinte:

```
switch (expressão-de-seleção) {  
  case <exp1>:  
    <sequência-de-comandos-1>  
    break;  
  case <exp2>:  
    <sequência-de-comandos-2>  
    break;  
  case <expn>:  
    <sequência-de-comandos-n>  
    break;  
  ...  
  default:  
    <sequência-de-comandos-extra>  
}
```

Assim que funciona:

1. A expressão **switch** é avaliada uma vez.
2. O valor da expressão é comparado com os valores de cada **case**.
3. Se houver uma correspondência, o bloco de código associado é executado.
4. Caso nenhum case seja válido, o bloco em **default** será executado, caso exista.

Veja o exemplo a seguir, que ilustra bem o que faz este comando:

```
let timefut: string = "";  
  
var readline = require('readline');  
  
var leitor = readline.createInterface({  
  input: process.stdin,  
  output: process.stdout  
});  
  
leitor.question("Digite o nome do time?\n", function(answer:string) {  
  var timefut = answer;  
  switch (timefut) {  
    case "Flamengo":  
    case "Fluminense":  
    case "Vasco":  
    case "Botafogo":  
      console.log("É um time carioca.");  
      break;  
    case "São Paulo":  
    case "Palmeiras":  
    case "Santos":  
    case "Corinthians":
```

```

        console.log("É um time paulista.");
        break;
    default:
        console.log("É de outro estado.");
    }
    leitor.close();
});

```

- 8) Escreva um script que atribua a uma variável “letra” uma letra do alfabeto. O script deverá informar ao usuário, se trata-se de uma vogal ou uma consoante.

1	Crie o script no seu projeto.	exercício08.ts
2	Crie uma variável para receber uma vogal e atribua o valor “a”.	var letra:string = “a”;
3	Utilizando o comando switch/case, verifique se a letra digitada é uma vogal ou consoante, mostrando no console a mensagem correspondente.	switch (letra) { case “a”: case “e”: case “i”: case “o”: case “u”: console.log (“A letra digitada é uma vogal!”); break ; default : console.log (“A letra digitada é uma consoante!”) }
4	Salve e execute seu script.	npm run ex08

- 9) Faça um script que solicite ao usuário que digite um número que represente um determinado mês do ano. Após a leitura escreva por extenso qual o mês lido. Caso o número digitado não esteja na faixa de 1 <= 12 escreva uma mensagem no console informando ao usuário do erro na digitação.

- 10) Uma empresa dará um aumento de salário aos seus funcionários de acordo com a categoria de cada empregado. O aumento seguirá a seguinte as seguintes regras:

- a) Categorias A, C, F, e H ganharão 10% de aumento sobre o salário
- b) Categorias B, D, E, I, J e T ganharão 15% de aumento sobre o salário
- c) Categorias K e R ganharão 25% de aumento sobre o salário
- d) Categorias L, M, N, O, P, Q e S ganharão 35% de aumento sobre o salário
- e) Categorias U, V, X, Y, W e Z ganharão 50% de aumento sobre o salário

Faça um script que solicite ao usuário que informe o nome, categoria e salário de um empregado. Seguindo a tabela acima, utilize a estrutura **switch/case** para calcular o reajuste correto e mostrar no console os dados informados do empregado mais o salário reajustado.