Condicionais

Desenvolver Algoritmos

Msc. Lucas G. F. Alves
e-mail: lgfalves@senacrs.com.br

Senac

Planejamento de Aula

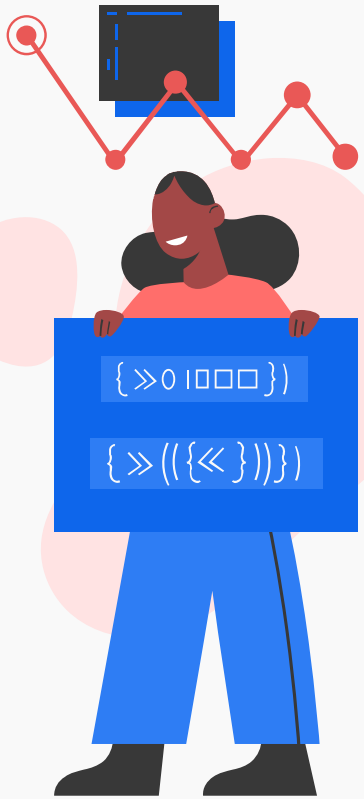
Revisão Exercícios

Revisão Comparadores

if e else

switch case

Exercícios



Revisão Exercícios

Exercícios de Fixação

Exercícios de interpretação de código

1. Indique todas as mensagens impressas no console, SEM EXECUTAR o programa.

//variável array será exibida como undefined
//variável array será exibida como null
//será exibido o número 11
//será exibido o valor da posicao 0 que no caso é 3
//será trocado o valor da posicao 1 para 19
//será salvo o valor da posição 6 e será exibido.

```
let array
console.log('a. ', array)

array = null
console.log('b. ', array)

array = [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
console.log('c. ', array.length)

let i = 0
console.log('d. ', array[i])

array[i+1] = 19
console.log('e. ', array)

const valor = array[i+6]
console.log('f. ', valor)
```

Exercícios de Fixação

Exercícios de interpretação de código

2. Leia o código abaixo com atenção

```
const frase = prompt("Digite uma frase")

console.log(frase.toUpperCase().replaceAll("A", "I"), frase.length)
```

Qual será o valor impresso no console se a entrada do usuário for: "Subi num ônibus em Marrocos"?

//a frase que será recebida será colocada toda em maiúscula e trocado as letras A por I, por fim será exibido a frase junto com seu tamanho.

A frase ficará: "SUBI NUM ÔNIBUS EM MIRROCOS"

Exercícios de Fixação

Exercícios de escrita de código

3. Faça um programa que pergunte ao usuário seu nome e seu e-mail. Em seguida, Imprima no console a seguinte mensagem:

O e-mail `emailDoUsuario` foi cadastrado com sucesso. Seja bem-vinda(o), `nomeDoUsuario`!

```
const nomeUsuario = prompt("Digite seu nome")
const emailUsuario = prompt("Digite seu email")
const frase = "O email " + emailUsuario + " foi cadastrado com sucesso. Seja bem-vinda(o) "+ nomeUsuario+ " !"
console.log(frase)
```

Exercícios de Fixação

Exercícios de escrita de código

4. Faça um programa que contenha um array com 5 das suas comidas preferidas, armazenado em uma variável. Em seguida, siga os passos:

const comidasPreferidas = [“pizza”, “lasanha”, “xis”, “carne”, “sushi”]

a) Imprima no console o array completo console.log(comidasPreferidas)

b) Imprima no console a mensagem "Essas são as minhas comidas preferidas: ", seguida por cada uma das comidas, **uma embaixo da outra**.
console.log(comidasPreferidas[0]) console.log(comidasPreferidas[1]) ...

c) Pergunte ao usuário uma comida preferida. Troque a segunda comida da sua lista pela inserida pelo usuário. Imprima no console a nova lista.

const comidaUsuario = prompt(“Digite sua comida”)
comidasPreferidas[1] = comidaUsuario console.log(comidasPreferidas)

Exercícios de Fixação

Exercícios de escrita de código

5. Faça um programa, seguindo os passos:

a) Crie um array vazio e guarde-o em uma variável, chamada `listaDeTarefas`. let listaDeTarefas = []

b) Pergunte ao usuário 3 tarefas que ele precise realizar no dia e armazene-as, uma por uma, no array. const tarefa1 = prompt("Digite a tarefa") .... listaDeTarefas.push(tarefa1)

c) Imprima o array no console. console.log(listaDeTarefas)

d) Peça ao usuário que digite o **índice** da tarefa que ele já realizou: 0, 1 ou 2. const indice = Number(prompt("Digite o número da tarefa realizada"))

e) Remova da lista o item de índice que o usuário escolheu.

f) Imprima o array. listaDeTarefas.splice(indice,1)/console.log(listaDeTarefas)

Desafios

1. Receba uma frase e retorne um array onde cada elemento é uma das palavras da frase, ignorando os espaços

2. Dado o array `["Banana", "Morango", "Abacaxi", "Laranja", "Ameixa"]`, faça um programa que acha o índice da palavra Abacaxi e imprime tanto o índice quanto o tamanho do array

Revisão Comparadores

Revisão Comparadores

Relembrando Comparadores

Comparadores são operadores que permitem **comparar** duas variáveis entre si.

O resultado destes operadores é sempre um **booleano.**

Quando a comparação for **correta**, o resultado é **true**. Caso contrário, **false.**

Revisão Comparadores

Relembrando Comparadores

- **===** : valor e tipo iguais;
- **!==** : valor ou tipo diferentes;
- **>** : maior que;
- **>=** : maior ou igual que;
- **<** : menor que;
- **<=** : menor ou igual que;

Revisão Comparadores

Relembrando Comparadores

- **===** : valor e tipo iguais;
- **!==** : valor ou tipo diferentes;
- **>** : maior que;
- **>=** : maior ou igual que;
- **<** : menor que;
- **<=** : menor ou igual que;

Condicionais

Condicionais

O que são?

Vocês já repararam quantas vezes ao dia temos que tomar uma decisão dependendo de outros fatores?

Nossos cérebros são incríveis e fazem isso de forma automática!

Vamos tentar pegar alguns exemplos e quebrar o processo de raciocínio em pedacinhos:

Condicionais

O que são?

Eu abro as cortinas para ver o dia lá fora...

- Se está chovendo:
    - Saio correndo para recolher as roupas do varal;
- Senão:
    - Fico de boa aproveitando o dia bonito;

Condicionais

O que são?

Testando um exercício da lista...

- Se passa no teste:
  - Fico feliz e vou pro próximo;
- Senão:
  - Choro e volto para achar o bug ;

Condicionais

O que são?

Estou ficando com fome e resolvo abrir a geladeira...

- Se tem alguma coisa além de gelo e ketchup:
  - Fico feliz por ter sido responsável;
  - Preparo um almoço gostoso;

- Senão:
  - Peço um hambúrguer por aplicativos ;

Condicionais

O que são?

Condicionais são estruturas de código usadas para fazer escolhas baseadas em alguns critérios.

Em outras palavras, elas permitem realizar uma determinada ação dependendo de uma condição.

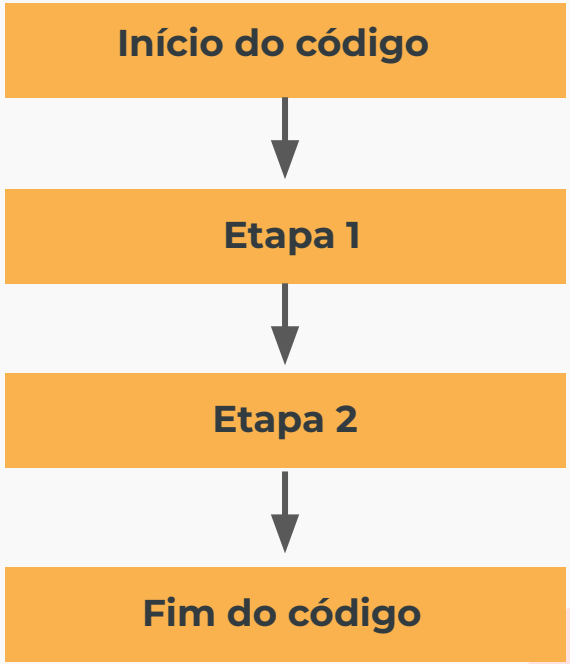
Exemplo: baseado na condição de estar chovendo eu vou realizar a ação de recolher a roupa.

Condicionais

Então

Até agora vimos que o javascript executa linha por linha de código, de forma síncrona e sequencial.

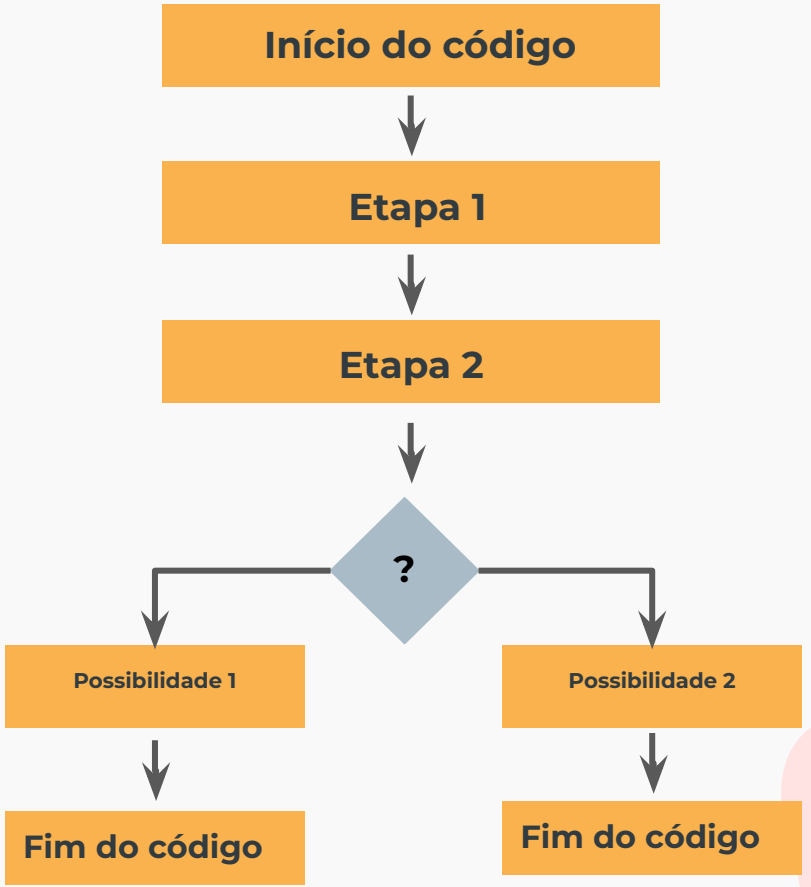
Como uma escada, que descemos degrau por degrau, sem poder pular nenhum.



Condicionais

Então

As condicionais são **estruturas de código** que nos dão o poder de **decidir** se a próxima linha de código deve ser executada ou não.



Condicionais

Definição de estrutura

Modo como alguma coisa é **construída**, **organizada** ou está disposta: a estrutura de uma empresa.

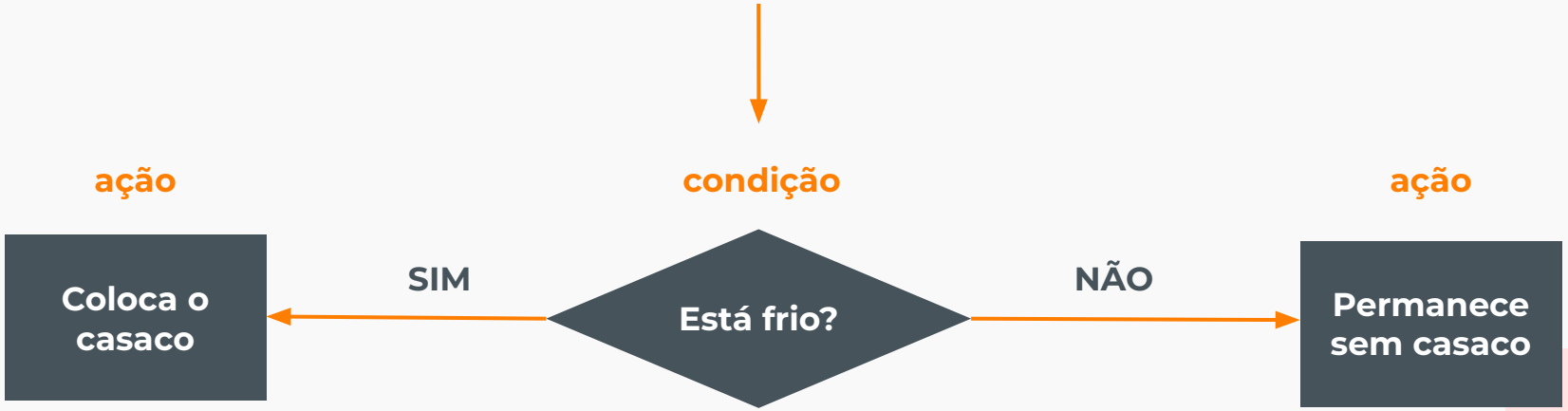
Aquilo que serve de **base** para algo; armação ou esqueleto: a estrutura de um edifício; a estrutura de uma linguagem de programação.

Árvores de Condicionais

Árvores de Condicionais

Como representar?

Uma maneira esquemática de representar condicionais é utilizando fluxogramas:



Árvores de Condicionais

Definição de estrutura

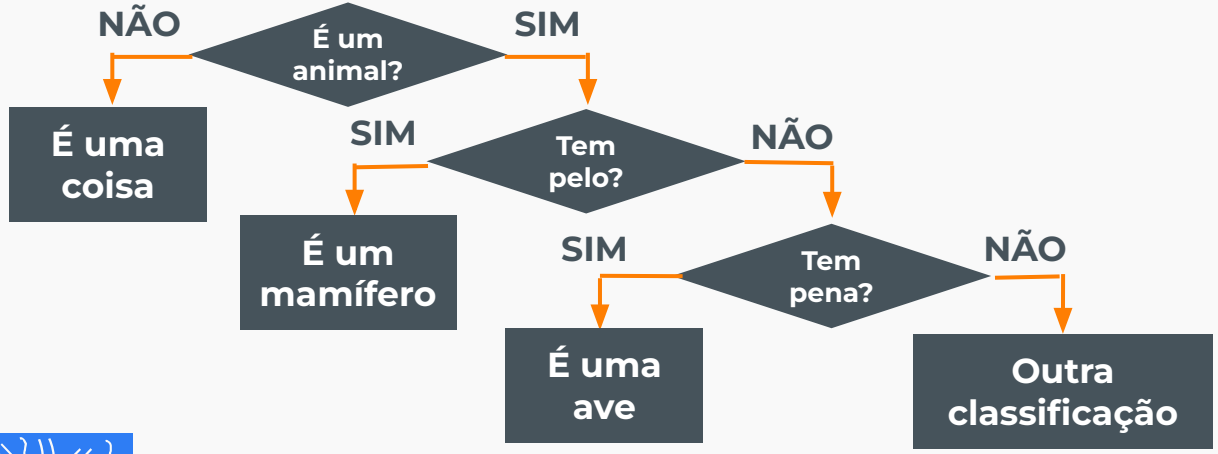
Nos fluxogramas:
Um losango representa uma **condição** para a **tomada de decisão.**

Um retângulo representa a **ação.**

Árvores de Condicionais

Definição de estrutura

Você pode **encadear** várias condições antes de chegar a uma resposta (por isso chamamos de árvore!)

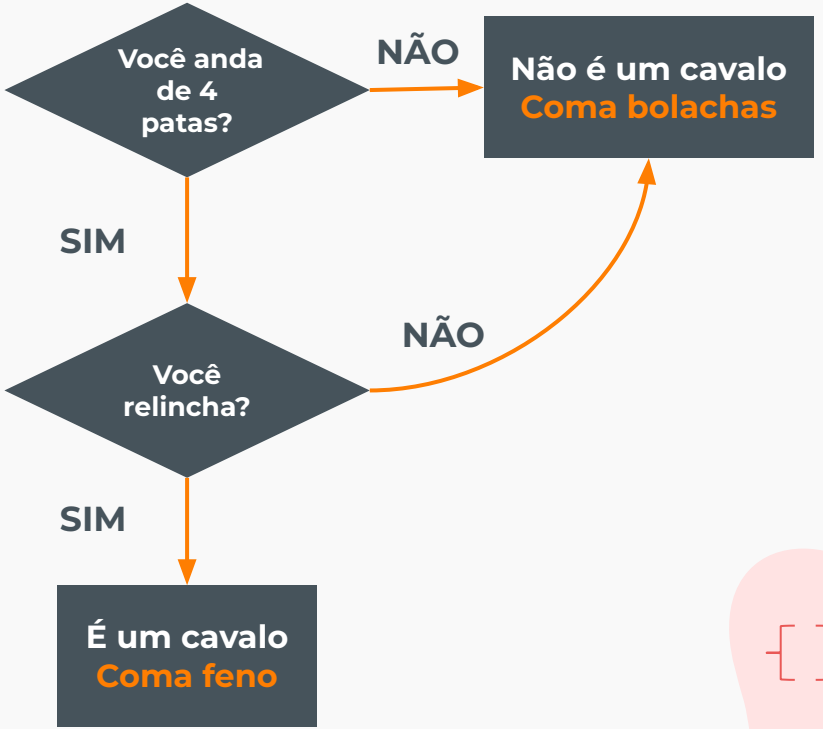


Árvores de Condicionais

Definição de estrutura

Depois de percorrer várias condições, você chega à uma conclusão

E, a partir da conclusão, você pode realizar alguma ação

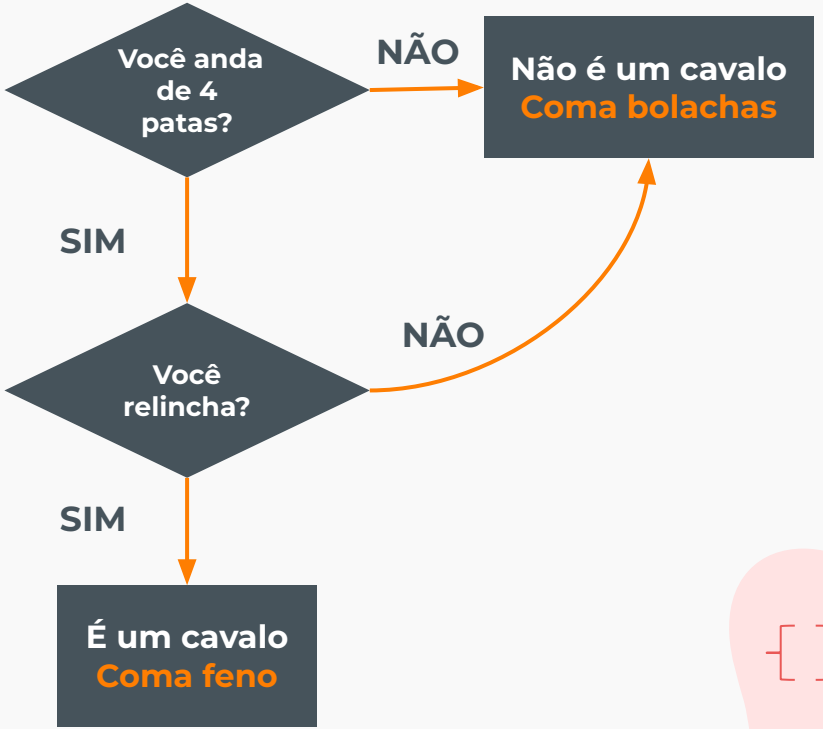


Árvores de Condicionais

Definição de estrutura

Depois de percorrer várias condições, você chega à uma conclusão

E, a partir da conclusão, você pode realizar alguma ação



Resumo Condicionais

Comparadores são operadores usados para comparar variáveis (===, !==, >, <).

Condicional é uma estrutura que permite escolher uma ação dependendo de uma condição.

Árvores de condicionais são uma maneira de se representar esquematicamente os condicionais.

Condicionais em JS

Condicionais em JS

Bloco if / else

if / else é a **sintaxe (estrutura)** de programação utilizada para **condicionais**.

Se a **condição for verdadeira** o código **dentro do if** é executado.

Condicionais em JS

if

Todo código da ação vai entre chaves { }

condição simples

```
let condicao1 = true

if (condicao1){
    // Como o valor da condição é true,
    // o código desse bloco é executado
    console.log('Entrei no if 1!')
}
```

```
let condicao2 = false

if (condicao2){
    // Como o valor da condição é false,
    // o código desse bloco NÃO é executado
    console.log('Entrei no if 2!')
}
```

({(({ ≫}))≪ })

Exercícios

1. Crie um programa que:

a) Recebe 2 números (chamaremos de num1 e num2)

b) Compara esses números entre si:

c) Se os números **forem iguais**, imprime uma mensagem de sucesso

Condicionais em JS

If + else

Todo código da ação vai entre chaves { }

```
let condicao = false

if (condicao){
    console.log('Entrei no if!')
} else {
    // Como o valor da condição é false,
    // o código do bloco else será executado
    console.log('Entrei no else!')
}
```

Exercícios

1. Crie um programa que:

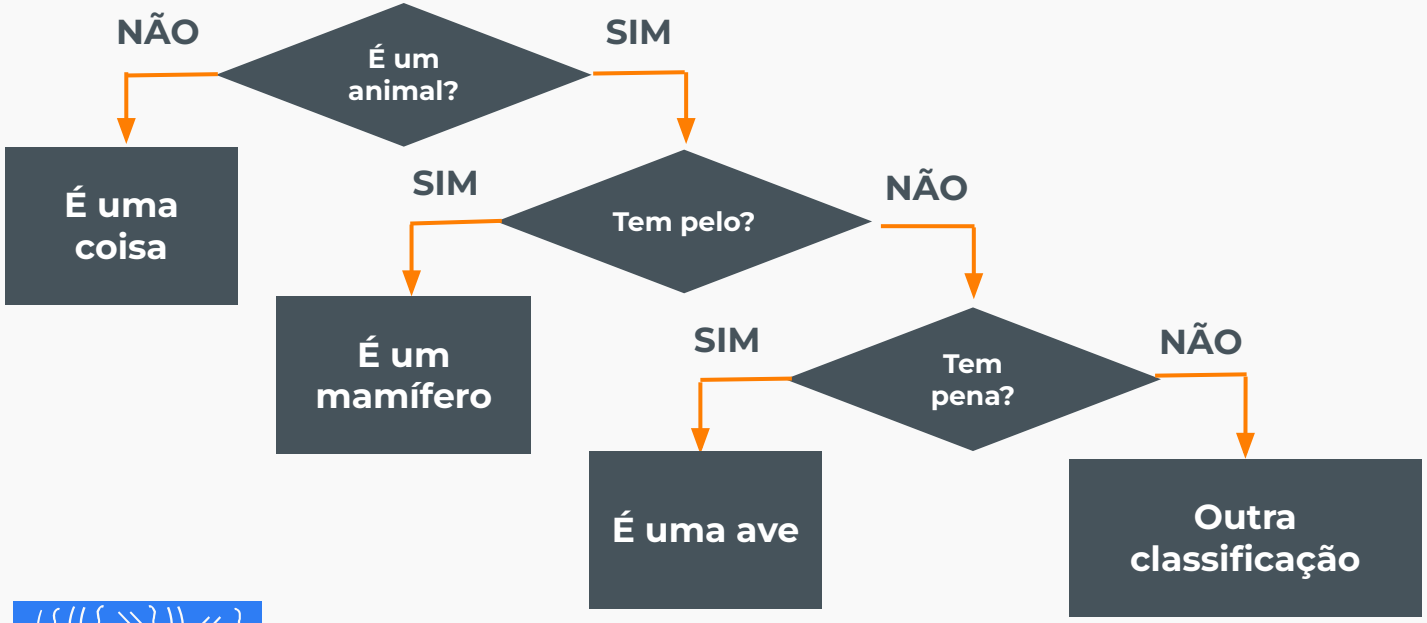
a) Recebe 2 números (chamaremos de num1 e num2)

b) Compara esses números entre si:

c) Imprime mensagens dizendo se os números são iguais ou diferentes.

Condicionais em JS

Encadeamento de condições



Condicionais em JS

if + else + if

Todo código da ação vai entre chaves { }

```
let condicao1 = false
let condicao2 = true

if (condicao1){
    console.log('Entrei no if 1!')
} else {
    // Como o valor da condicao1 é false,
    // o código do else será executado
    if (condicao2) {
        // Como o valor da condicao2 é true,
        // executaremos esse código!
        console.log('Entrei no if 2!')
    }
}
```

Condicionais em JS

if + else if

```
let condicao1 = false
let condicao2 = true

if (condicao1){
    console.log('Entrei no if 1!')
} else {
    if (condicao2) {
        console.log('Entrei no if 2!')
    }
}
```

```
let condicao1 = false
let condicao2 = true

if (condicao1){
    console.log('Entrei no if 1!')
} else if (condicao2) {
    console.log('Entrei no if 2!')
}
```

Condicionais em JS

if + else if + else

```
let condicao1 = false
let condicao2 = false

if (condicao1){
    console.log('Entrei no if 1!')
} else if (condicao2) {
    console.log('Entrei no if 2!')
} else {
    // Como tanto o valor da condicao1 e
    // da condicao2 são false, executa
    // os comandos do bloco else
    console.log('Entrei no else!')
}
```

Exercícios

1. Crie um programa que:

a) Recebe 2 números (chamaremos de num1 e num2)

b) Compara esses números entre si:

c) Imprime mensagens dizendo se o primeiro número é **maior/menor/igual** ao segundo número.

Resumo

Vimos como a gente pode pedir para o computador realizar condições usando o JavaScript:

Condicional simples (**if**)

Condicional composta (**if/else**)

Condicional aninhada (**if + else if + else**)

Switch Case

Switch Case

Onde usar?

Vamos dar uma olhada no código:

```
let paisDeOrigem
if (paisDeOrigem === 'Brasil'){
    console.log('brasileiro')
} else if (paisDeOrigem === 'EUA'){
    console.log('norte americano')
} else if (paisDeOrigem === 'Inglaterra'){
    console.log('inglês')
} else if (paisDeOrigem === 'França'){
    console.log('francês')
} else if (paisDeOrigem === 'Itália'){
    console.log('italiano')
} else if (paisDeOrigem === 'Canadá'){
    console.log('canadense')
} else {
    console.log('nacionalidade não encontrada')
}
```

Switch Case

Comparando com if/else

```
let paisDeOrigem
if (paisDeOrigem === 'Brasil'){
    console.log('brasileiro')
} else if (paisDeOrigem === 'EUA'){
    console.log('norte americano')
} else if (paisDeOrigem === 'Inglaterra'){
    console.log('inglês')
} else if (paisDeOrigem === 'França'){
    console.log('francês')
} else if (paisDeOrigem === 'Itália'){
    console.log('italiano')
} else if (paisDeOrigem === 'Canadá'){
    console.log('canadense')
} else {
    console.log('nacionalidade não encontrada')
}
```

```
let paisDeOrigem
switch (paisDeOrigem){
    case 'Brasil':
        console.log('brasileiro')
        break
    case 'EUA':
        console.log('norte americano')
        break
    case 'Inglaterra':
        console.log('inglês')
        break
    default:
        console.log('nacionalidade não encontrada')
        break
}
```

Switch Case

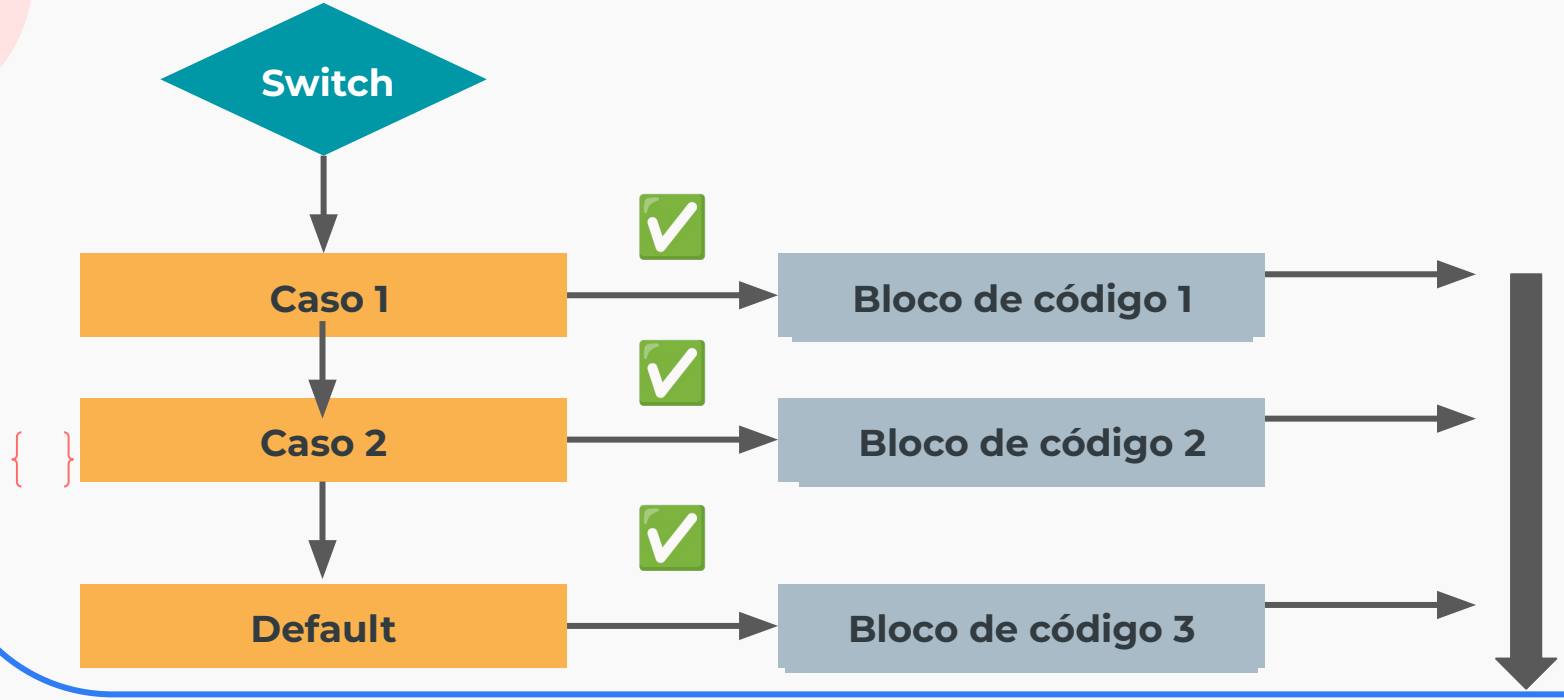
Estrutura

```
let paisDeOrigem
if (paisDeOrigem === 'Brasil'){
    console.log('brasileiro')
} else if (paisDeOrigem === 'EUA'){
    console.log('norte americano')
} else if (paisDeOrigem === 'Inglaterra'){
    console.log('inglês')
} else if (paisDeOrigem === 'França'){
    console.log('francês')
} else if (paisDeOrigem === 'Itália'){
    console.log('italiano')
} else if (paisDeOrigem === 'Canadá'){
    console.log('canadense')
} else {
    console.log('nacionalidade não encontrada')
}
```

```
let paisDeOrigem
switch (paisDeOrigem){
    case 'Brasil':
        console.log('brasileiro')
        break
    case 'EUA':
        console.log('norte americano')
        break
    case 'Inglaterra':
        console.log('inglês')
        break
    default:
        console.log('nacionalidade não encontrada')
        break
}
```

Switch Case

Estrutura



Switch Case

Estrutura

```
let paisDeOrigem
switch (paisDeOrigem){
    case 'Brasil':
        console.log('brasileiro')
        break
    case 'EUA':
        console.log('norte americano')
        break
    case 'Inglaterra':
        console.log('inglês')
        break
    default:
        console.log('nacionalidade não encontrada')
        break
}
```

Conseguimos colocar um caso padrão chamado **default**

O código dentro dele será executado se o valor da variável **não bater com as opções** dos **cases**

Switch Case

Estrutura

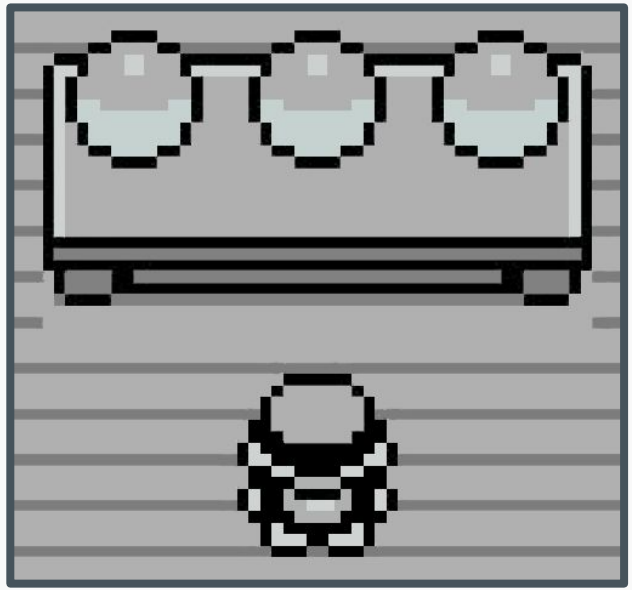
```
let paisDeOrigem
switch (paisDeOrigem){
    case 'Brasil':
        console.log('brasileiro')
        break
    case 'EUA':
        console.log('norte americano')
        break
    case 'Inglaterra':
        console.log('inglês')
        break
    default:
        console.log('nacionalidade não encontrada')
        break
}
```

break é a palavra que faz com que a execução do código saia do bloco em questão. Caso não exista o break, o código continuará executando

Exercícios

Escreva um programa que receba o Pokémon inicial escolhido pela pessoa e imprima no console o seu tipo:

- Bulbasaur (Planta e Veneno);
- Charmander (Fogo);
- Squirtle (Água);



Exercícios

Escreva o programa do chapéu seletor que recebe a aluna e imprima no console a sua casa de Hogwarts:

- Hermione(Grifinória);
- Ana(Lufa-lufa);
- Luna(Corvinal);
- Narcisa(Sonserina)

Resumo

Podemos realizar condições usando no JavaScript o **if + else**. Além disso, podemos usar o switch case.

- Switch case evita códigos verbosos;
- Só serve para verificar casos de igualdade;
- Permite executar bloco de código diferente baseado em cada opção;
- Break = Para a execução do código quando a condição é atendida;

Condicionais e Operadores Lógicos

Condicionais e Operadores Lógicos

Relembrando Operadores Lógicos

- && : AND - verdadeiro se ambos os operandos forem verdadeiros.
- || : OR - falso se ambos os operadores forem falsos.
- ! : NOT negação - sua função é simplesmente inverter os valores.

Condicionais e Operadores Lógicos

Como os **operadores lógicos** retornam booleanos, conseguimos usá-los **diretamente** na condição dos blocos if/else

```
let condicao1 = true
let condicao2 = false

if (condicao1 && condicao2){
    // Entra aqui se ambas forem true
}

if (!condicao1){
    // Entra aqui se condicao1 for false
}
```

Exercícios

Uma pessoa pode estudar em uma faculdade se:

- Tiver concluído o ensino médio;
- Tiver 18 anos ou mais;
- Não estiver cursando outra faculdade;

Escreva um programa que receba estes parâmetros e imprima se a pessoa pode ou não estudar nesta faculdade. Receba os dados do usuário pelo prompt.

Desafio

Crie um **jogo de "Pedra, Papel, Tesoura, Lagarto, Spock"** no qual o usuário escolhe uma das opções, e o computador escolhe aleatoriamente outra. Utilize um **switch-case** para determinar o vencedor.

O jogo segue estas **regras**:Pedra esmaga Tesoura. Pedra esmaga Lagarto.Papel cobre Pedra. Papel desqualifica Spock.Tesoura corta Papel. Tesoura decapita Lagarto. Lagarto envenena Spock. Lagarto come Papel.Spock vaporiza Pedra. Spock quebra Tesoura.

Gere uma escolha aleatória para o computador:

```
const escolhaComputador = choices[Math.floor(Math.random() * 5)];
```

Use um switch-case para comparar as escolhas e determinar o vencedor.

Utilize operadores lógicos para definir as regras de vitória.

Resumo

Operadores de comparação são usados para se compararem valores de mais de uma variável. Eles sempre **retornam** um valor **booleano.**

- ===: valor e tipo iguais
- !==: valor ou tipo diferentes
- >: maior que
- >=: maior ou igual que
- <: menor que
- <=: menor ou igual que

Resumo

Condicionais são estruturas que simbolizam decisões tomadas dependendo de certas **condições.**

Árvore condicional é uma estrutura **esquemática** que pode ser usada para **facilitar** nossa análise e construção de condicionais

Resumo

if/else são blocos que permitem fazer uma condicional Eles recebem uma condição e o código:

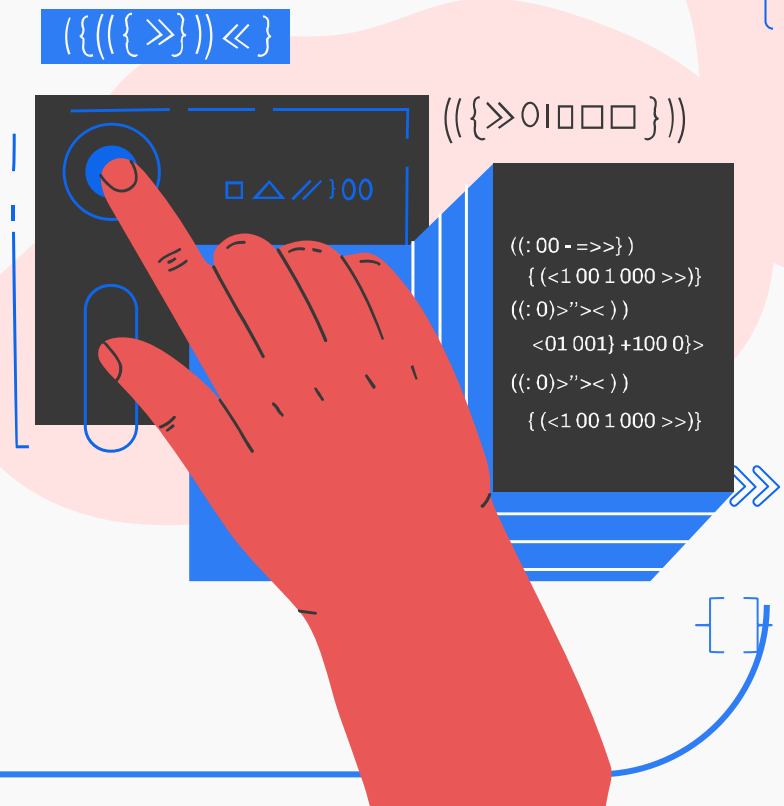
- do **if** é executado se a **condição** for **true;**
- do **else** é executado se a **condição** for **false;**

switch case são blocos que permitem simplificar **if/else** apenas no caso de **comparador de igualdade.**

- lembrem-se de escrever o break;

Obrigado!

E-mail: lgfalves@senacrs.com.br



Professor

Lucas G. F. Alves

Senac