

CAPACITAÇÃO EM 'JAVASCRIPT' {

[DIA 02] 

< PET INFORMÁTICA - UNIVERSIDADE ESTADUAL
DE MARINGÁ >

}

Roteiro 'Dia 02' {

01 Condições (if, else if, else e switch case)

< ... >

02 Arranjo e objetos

< ... >

03 Laço de repetição
(for, while e do while)

< ... >

}

Condicional: if e else

- Executa um bloco de código se EXPRESSÃO for verdadeira
- Caso contrário executa o segundo bloco

```
if (EXPRESSAO1) {  
    //EXPRESSAO1 é verdadeira  
} else {  
    //EXPRESSAO1 é falsa  
}
```

Condicional: if, else if e else

```
1
2
3     if (EXPRESSAO1) {
4         //EXPRESSAO1 é verdadeira
5     } else if (EXPRESSAO2) {
6         //EXPRESSAO2 é verdadeira
7     } else if (EXPRESSAO3) {
8         //EXPRESSAO3 é verdadeira
9     } else {
10        //nenhuma das expressões são verdadeiras
11    }
12
13
14
```

Condicional: Switch case

```
1
2      switch (op) {
3          case 1:
4              //executa quando op vale 1
5              break;
6          case 2:
7              //executa quando op vale 2
8              break;
9          case 3:
10             //executa quando op vale 3
11             break;
12         default:
13             //executa quando op vale algo diferente de 1, 2 ou 3
14             break;
15     }
```

Exercícios{

01 Bhaskara

Faça um algoritmo que classifique as raízes de uma equação de segundo grau com base no delta.

$$x = \frac{-b \pm \sqrt{\Delta}}{2a}$$

$$\Delta = b^2 - 4ac$$

02 Par ou Ímpar??

Faça um algoritmo que receba um número inteiro e verifique se esse número é par ou ímpar.

03 São Múltiplos?

Faça um algoritmo que receba dois números e determine se esses números são ou não múltiplos.

}

Arranjos / Array

- Um arranjo/array é uma forma de armazenar diversos valores em somente uma variável.
- Podemos criar um vetor/array e já atribuir valores em suas posições.

```
const frutas = ["Maçã", "Banana"];
```

```
const nums = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0];
```

```
const pessoa = ["Márcio", "Victor", 42];
```

Arranjos / Array

```
const nums = [];  
nums[0] = 1;  
nums[1] = 2;  
nums[2] = 3;  
nums[3] = 4;  
nums[4] = 5;  
nums[5] = 6;  
nums[6] = 7;  
nums[7] = 8;  
nums[8] = 9;  
nums[9] = 0;
```


Métodos de Array

Arrays são objetos semelhantes a listas que vêm com uma série de métodos embutidos para realizar operações de travessia e mutação

- Obtendo o tamanho de um array

```
const nums = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0];
```

```
console.log(nums.length); //imprime 10
```

- Ordenando um array, pode ou não ser numérico

```
1
2
3
4
5 const nums = [9, 2, 7, 6, 4, 3, 0, 7, 5, 6, 4];
6
7
8 nums.sort();
9
10 // [0, 2, 3, 4, 4, 5, 6, 6, 7, 7, 9]
11
12
13
14
```

- Transformando um array em uma string

```
1
2
3
4
5 const frutas = ["Banana", "Laranja", "Maça"];
6
7
8 const x = frutas.toString();
9
10 console.log(x); //Banana,Laranja,Maça
11
12
13
14
```

- Transformando um array em uma string com separador específico

```
1
2
3
4
5 const frutas = ["Banana", "Laranja", "Maça"];
6
7
8 const x = frutas.join("|");
9
10 console.log(x); //Banana|Laranja|Maça
11
12
13
14
```

```
1  const nums = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0];
```

- Acessar um item (index) do array

```
5  const primeiro = nums[0];
```

```
6  const ultimo = nums[nums.length - 1];
```

- Percorrendo todo o array

```
10 for (let i = 0; i < nums.length; i++) {  
11     console.log(nums[i]);  
12 }  
13  
14
```

- Adicionando e Excluindo um item no final do Array

```
1
2
3
4 let frutas = ["Maçã", "Banana"];
5 frutas.push("Laranja"); //retorna o índice do elemento
6 adicionado
7
8 //["Maçã", "Banana", "Laranja"]
9
10 frutas.pop(); //retorna o elemento que foi removido
11 //["Maçã", "Banana"]
12
13
14
```

- Removendo e adicionando um item no começo do Array

```
let frutas = ["Maçã", "Banana"];
```

```
frutas.shift(); //retorna o elemento que foi removido  
//["Banana"]
```

```
frutas.unshift("Morango"); //retorna o índice do elemento  
adicionado  
//["Morango", "Banana"]
```


- Concatenando dois arrays

```
const frutas1 = ["Banana", "Laranja", "Maça"];
```

```
const frutas2 = ["Limão", "Manga"];
```

```
const frutas = frutas1.concat(frutas2);
```

```
console.log(frutas);
```

```
//[Banana, Laranja, Maça, Limão, Manga]
```

- Altera elementos do array

```
const frutas = ["Banana", "Laranja", "Maça",  
"Manga"];
```

```
frutas.splice(2, 2, "Limão", "Kiwi");
```

```
console.log(frutas);
```

```
// [Banana, Laranja, Limão, Kiwi]
```

- Altera elementos do array

```
const frutas = ["Banana", "Laranja", "Maça",  
"Manga"];
```

```
frutas.splice(0, 1);
```

```
console.log(frutas);
```

```
// [Laranja, Maça, Manga]
```

- Cria uma cópia de parte de um array a partir de um subarray criado entre as posições início e fim

```
const frutas = ["Banana", "Laranja", "Limão",  
"Manga"];
```

```
const citricas = frutas.slice(1, 3);
```

```
console.log(citricas);
```

```
// [Laranja, Limão]
```

Objeto

- Um objeto é uma coleção de dados e/ou funcionalidades relacionadas (que geralmente consistem em diversas variáveis e funções – que são chamadas de propriedades e métodos quando estão dentro de objetos).

```
var pessoa = {  
  nome: "Márcio",  
  sobrenome: "Victor",  
  idade: 42,  
  sexo: "masculino",  
  interesses: ["música", "praia"],  
};
```

- Em JavaScript, um objeto é uma entidade independente, com propriedades e tipos.
- As propriedades de um objeto definem as características do objeto. Você acessa as propriedades de um objeto com uma simples notação de ponto:

```
console.log(pessoa.nome); // "Márcio"
console.log(pessoa.sobrenome); // "Victor"
console.log(pessoa.idade); // 42
console.log(pessoa.interesses[0]); // "música"
```

Laço de Repetição: For

- Realiza uma operação inicial
- Executa o bloco enquanto a expressão for verdadeira
- Realiza uma operação ao fim de cada execução

Laço de Repetição: For

```
1  
2  
3  
4  
5  
6     for(let i = 0; i < 10; i++) {  
7         //executa o bloco 10 vezes  
8     }  
9  
10  
11  
12  
13  
14
```


Laço de Repetição: For In

- Percorre as propriedade enumeradas de um objeto

```
1 let texto = "";
2
3
4
5
6
7 for (let i in pessoa) {
8     texto += pessoa[i] + " ";
9 }
10
11 console.log(texto);
12
13
14 // Márcio Victor 42 masculino música,praia
```

Laço de Repetição: For Of

- Percorre objetos iterativos (incluindo arrays)

```
const nums = [5, 7, 6, 2, 0, 8, 3, 7, 9];
```

```
let soma = 0;
```

```
for (let i of nums) {
```

```
    soma += i;
```

```
}
```

```
console.log(soma); //a soma de todo vetor (47)
```

Laço de Repetição: While

- Executa enquanto a expressão for verdadeira

```
let i = 0;
while(i < 10) {
    //executa o bloco 11 vezes
    i++;
}
```

Laço de Repetição: Do While

- Executa o bloco uma vez
- Depois enquanto a expressão for verdadeira o bloco é executado

```
let i = 0;  
do {  
    //executa o bloco 10 vezes  
    i++;  
} while(i < 10);
```

Break

- Interrompe o laço

```
1  const nums = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
2
3  //           0  1  2  3  4  5  6  7  8  9
4
5  for (let i = 0; i < nums.length; i++) {
6
7      console.log(nums[i]);
8
9      if (i > 4) {
10         break;
11     }
12 }
13
14 //imprime todos até o 5
```

Continue

- Interrompe a iteração e continua na próxima

```
1  const nums = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
2
3  //           0  1  2  3  4  5  6  7  8  9
4
5  for (let i = 0; i < nums.length; i++) {
6
7      if (i === 4) {
8          continue;
9      }
10     console.log(nums[i]);
11 }
12 //imprime todos menos o 4, que está na posição 4
```

Exercícios{

01 Maior e menor elemento

Faça um algoritmo que encontre o maior e o menor elemento em um array de n elementos.

02 Vetor de nomes

Crie um vetor de nomes e encontre seu nome utilizando algum laço de repetição

}

Obrigada(o) pela presença 🏆;

{

‘Dúvidas?’

pet@din.uem.br

@petinfouem

Comissão organizadora: Diogo, Nayane e Yuri +
Danilo, Gabriel, João Staub e Rold



Instagram

Referências Bibliográficas: w3schools - JavaScript



Facebook

CREDITS: This presentation template was
created by **Slidesgo**, including icons by
Flaticon, and infographics & images by **Freepik**

< Please keep this slide for attribution >

}