

INSTITUTO FEDERAL
Sul-rio-grandense

Câmpus
Charqueadas

EDUCAÇÃO
PÚBLICA
100%
GRATUITA

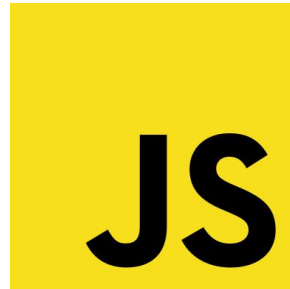
Programação Web I

Estruturas Homogêneas - Vetores e Matrizes

Prof. Fábio Luís da Silva Santos
fabiosantos@ifsul.edu.br

Estruturas Homogêneas

Vetores e Matrizes



- O que são Variáveis?
- Quais os tipos de dados que podem ser armazenados?

Estruturas Homogêneas

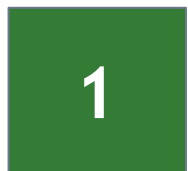
Vetores e Matrizes

- Estrutura de dados Homogênea ou Vetor/Matriz é uma coleção de variáveis de mesmo tipo ou de tipos diferentes, acessíveis com um único nome e armazenados contiguamente (um após o outro) na memória.
- O acesso de forma individualizada a cada elemento, seja para leitura ou escrita, é realizada através de índices.

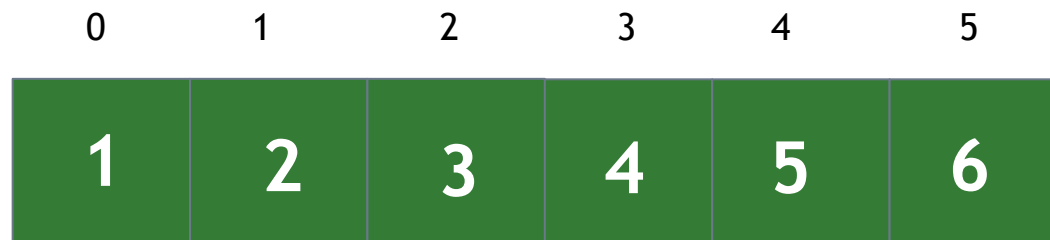
Estruturas Homogêneas

Vetores e Matrizes

- Vetores
 - Estruturas com apenas uma dimensão



```
var num = 1;
```



```
var numeros = [1,2,3,4,5,6];
```

Estruturas Homogêneas

Vetores e Matrizes

- Acessando as posições do vetor

0	1	2	3	4	5
1	2	30	4	50	6

```
var numeros = [1,2,3,4,5,6];
```

```
numeros[2] = 30;
```

```
numeros[4] = 50;
```

Estruturas Homogêneas

Vetores e Matrizes

- Acessando as posições do vetor

0	1	2	3	4	5
1	2	3	4	5	6

```
var numeros = [1,2,3,4,5,6];
```

`numeros[3]`

`numeros[5]`

Estruturas Homogêneas

Vetores e Matrizes

0	1	2	3	4	5
Internacional	Grêmio	Cruzeiro	Caxias	Ipiranga	Brasil de Pelotas

```
var times = ['Internacional','Grêmio','Cruzeiro','Caxias','Ipiranga','Brasil de Pelotas'];
```

Estruturas Homogêneas

Vetores e Matrizes

- Acessando as posições do vetor

0	1	2	3	4	5
Internacional	Grêmio	Cruzeiro	Caxias	Ipiranga	Brasil de Pelotas

```
times[4]  
times[5]
```


#BoraCodificar

Estruturas Homogêneas

Vetores e Matrizes

- **exemplo01.html:**
 - **Declaração de variáveis - Revisão.**

```
<script>
    var numero = 1;
    document.write("variável numero: " + numero + "<br>");
    var letra = 'A';
    document.write("variável letra: " + letra + "<br>");
</script>
```

Estruturas Homogêneas

Vetores e Matrizes

- **exemplo02.html:**
 - **Estruturas Homogêneas - Declarando um vetor de Inteiros**

```
<script>
  var numeros = [1,2,3,4,5,6];
  document.write(numeros + ' Conteúdo do vetor números...<br>');
  document.write(numeros.length + " é o número de elementos do vetor! <br>");
</script>
```

Estruturas Homogêneas

Vetores e Matrizes

- **exemplo03.html:**
 - **Estruturas Homogêneas - Define um Vetor, depois lê 5 números inteiros via prompt.**

```
<script>
    var valores = [];

    for(var i = 0; i<5; i++){
        valores[i] = window.prompt("Digite um número inteiro: ");
    }

    document.write(valores);

</script>
```

Estruturas Homogêneas

Vetores e Matrizes

- **exemplo04.html:**
 - **Estruturas Homogêneas - Define um Vetor de Strings.**

```
<script>
var times = ["Internacional","Grêmio","Cruzeiro","Caxias","Ipiranga","Brasil"];
document.write(times + "<br>");
document.write("O número de elementos do vetor times é: " + times.length);
console.table(times); // mostra no console na forma de tabela...
</script>
```

#AgoraComVocês

Estruturas Homogêneas

Vetores e Matrizes

- **exercício01.html:**
 - Crie um script, defina um vetor chamado elementos, faça a leitura de 10 números inteiros via `window.prompt()` armazenando os valores no vetor. Depois exiba o conteúdo do vetor elementos.

Estruturas Homogêneas

Vetores e Matrizes

- exercício02.html:
 - Crie um script, defina um vetor chamado elementos, faça a leitura de 10 números inteiros via `window.prompt()` armazenando os valores no vetor. Calcule a soma de todos os valores digitados. Depois exiba o conteúdo do vetor elementos mais a soma dos elementos.

#fato



Estruturas Homogêneas

Vetores e Matrizes

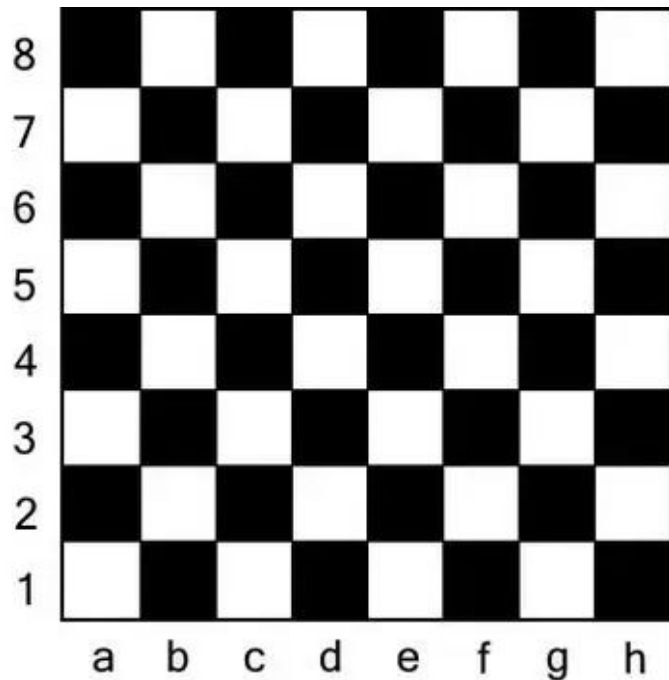
- **exercício03.html:**
 - Crie um script, defina um vetor chamado elementos, faça a leitura de 10 números inteiros via `window.prompt()` armazenando os valores no vetor. Calcule a soma de todos os valores digitados. Depois exiba apenas os valores pares contidos no vetor.

#Voltamos

Estruturas Homogêneas

Vetores e Matrizes

- Matrizes
 - Estruturas com mais de uma dimensão



	A	B	C	D	E	F	G	H
1	Nome	Sobrenome	Nome Completo	Data de Nascimento	Estado de Nascimento	DDD	Celular	CPF
1961	Thaís	Nemitz	Thaís Nemitz	15/01/1988	Amazonas (AM)	92	905546617	57647734950
1962	Saranna	Brasil	Saranna Brasil	01/06/1986	Mato Grosso (MT)	65	977819627	13208277890
1963	Flávia	Moss	Flávia Moss	23/10/1988	Amazonas (AM)	92	907492030	62415207369
1964	Thárcila	Cravo	Thárcila Cravo	10/02/1991	Paraná (PR)	41	962987521	83753019082
1965	Yasmini	Lana	Yasmini Lana	15/05/1992	Sergipe (SE)	79	970366771	62779432904
1966	Cleo	Areias	Cleo Areias	13/10/1993	Mato Grosso do Sul (MS)	84	938795706	6091984159
1967	Frederico	Rozenbaum	Frederico Rozenbaum	07/09/1987	Rio Grande do Sul (RS)	51	934662145	8476561670
1968	Priscilla	Caldana	Priscilla Caldana	27/04/1993	Amazonas (AM)	92	987555854	54545699165
1969	Cristina	Priscilla	Cristina Priscilla	21/03/1988	Pernambuco (PE)	81	903611779	80073034454
1970	Lohany	Cesar	Lohany Cesar	03/10/1992	Mato Grosso (MT)	65	909510294	19455987808
1971	Fabiane	Guilherme	Fabiane Guilherme	31/01/1991	Pará (PA)	91	937316265	37695956499
1972	Adrian	Torres	Adrian Torres	13/07/1993	São Paulo (SP)	11	922715385	31646626304

#BoraCodificar

Estruturas Homogêneas

Vetores e Matrizes

- exemplo05.html
 - Boletim Escolar

	Matemática (Coluna 0)	PWI (Coluna 1)	Banco de Dados (Coluna 2)
Etapa 1 (Linha 0)	9	7	6,5
Etapa 2 (Linha 1)	9,8	9	8,5

```
var boletim = [[9,7,6.5],[9.8,9,8.5]];
```

Estruturas Homogêneas

Vetores e Matrizes

- exemplo05.html
 - Boletim Escolar

	Matemática (Coluna 0)	PWI (Coluna 1)	Banco de Dados (Coluna 2)
Etapa 1 (Linha 0)	9	7	6,5
Etapa 2 (Linha 1)	9,8	9	8,5

```
document.write(boletim[1][2]);  
//primeiro colchete endereça a linha  
//segundo colchete endereça a coluna
```

Estruturas Homogêneas

Vetores e Matrizes

- exemplo05.html
 - Boletim Escolar

```
boletim[0][1] = 10;
```

	Matemática (Coluna 0)	PWI (Coluna 1)	Banco de Dados (Coluna 2)
Etapa 1 (Linha 0)	9	10	6,5
Etapa 2 (Linha 1)	9,8	9	8,5

Estruturas Homogêneas

Vetores e Matrizes

- exemplo05.html
 - Boletim Escolar

	Matemática (Coluna 0)	PWI (Coluna 1)	Banco de Dados (Coluna 2)
Etapa 1 (Linha 0)	9	10	6,5
Etapa 2 (Linha 1)	9,8	9	8,5

// imprime apenas as informações da primeira linha (0)

```
for(var c = 0; c <= 2; c++){  
    document.write(boletim[0][c] + ' - ' );  
}
```


#AgoraComVocês

Estruturas Homogêneas

Vetores e Matrizes

- exercício04.html:
 - crie uma matriz chamada tabela com a seguinte composição de valores

20	30	40	50
100	200	300	400
1	2	2	4

- utilizando `console.table()` mostre os valores contidos na matriz

Programar
é a arte exata de
fazer algo que nunca
exatamente
está pronto!



Piadas de Programador

#fato

Estruturas Homogêneas

Vetores e Matrizes

- **exercício05.html:**
 - **crie uma matriz chamada tabela com a seguinte composição de valores**

1	10	20	30
2	11	21	31
3	12	22	32

- **agora mostre somente os valores da linha 0, utilize para isso uma estrutura de repetição**

Estruturas Homogêneas

Vetores e Matrizes

- **exercício06.html:**
 - lançamento de um dado - crie um script que contenha um vetor chamado dado com os valores 1, 2, 3, 4, 5 e 6. Depois faça um sorteio entre os índices do vetor com o objetivo de imprimir o valor correspondente ao índice sorteado. Para sortear um índice do vetor dado, utilize

```
var nro = Math.floor(Math.random() * 6); // 6 faces do dado  
document.write('<br>' + nro);
```

#MétodosUteis

Arrays, Matrizes e Objetos

Arrays

Funções úteis para usar com arrays

- Push: Insere elementos no fim do array
- indexOf: Procura um valor e retorna o índice.
- Slice: Remove elementos do array

```
const arr = [];  
arr.push(5);  
arr.push(1, 2, 3.1415, 4, 'texto');  
console.log(arr);  
// [5,1,2,3.1415,4,"texto"]
```

```
const arr = [10,20,30,40,50,60,70,80,90];  
console.log(arr.indexOf(40)); // 3  
console.log(arr.indexOf(100)); // -1
```

```
const arr = [10,20,30,40,50];  
console.log(arr.slice(2));  
// [30,40,50]
```

```
const arr = [10,20,30,40,50];  
console.log(arr.slice(1,4));  
// [20,30,40]
```

```
const arr = [10,20,30,40,50];  
console.log(arr.slice(-3));  
// [30,40,50]
```

Arrays, Matrizes e Objetos

Arrays

exercício07.html - Parte 1:

- Gere 100 número aleatórios entre 0 e 1000.
 - Use a função random ao lado.
- Insira os números gerados em um array usando o **push**.
- Remova do array os 5 primeiros e os 5 últimos números.
 - Use **slice** para remover os números.
- Mostre o conteúdo do array no console.

```
// número entre 0 e 10  
const num = Math.random() * 10;
```

```
< (90) [729, 174, 29, 638, 145, 892, 105, 379, 508, 66, 961, 245, 88  
6, 84, 821, 34, 75, 9, 632, 21, 665, 696, 567, 436, 321, 621, 749,  
326, 688, 849, 303, 866, 666, 186, 526, 954, 516, 803, 91, 722, 6  
▶ 4, 563, 276, 126, 947, 222, 466, 712, 116, 354, 788, 214, 571, 69  
5, 175, 904, 296, 821, 104, 501, 895, 3, 574, 93, 394, 432, 805, 6  
76, 999, 426, 605, 917, 941, 541, 34, 176, 468, 138, 264, 277, 57  
5, 504, 765, 567, 661, 845, 562, 916, 86, 275]
```

Arrays, Matrizes e Objetos

Arrays

Funções úteis para usar com arrays

- Split: Divide uma string em arrays.
- Join: Junta um array transformando em uma string.
- Sort: ordena os elementos do array.

```
const text = 'Cada palavra está separada por um espaço';  
const words = text.split(' ');  
console.log(words);  
// ["Cada","palavra","está","separada","por","um","espaço"]
```

```
const data = [1,2,3,4,5,6,7];  
const text = `

${ data.join('</div><div>') }</div>`;  
console.log(text);  
//  
"<div>1</div><div>2</div><div>3</div><div>4</div><div>5</div>  
<div>6</div><div>7</div>"


```

```
let arr = [6,4,3,7,3,10,8,11,9,2];  
arr = arr.sort((a,b) => a-b);  
console.log(arr); // [2,3,3,4,6,7,8,9,10,11]
```


Arrays, Matrizes e Objetos

Arrays

exercício07.html - Parte 2:

- Use o array do exercício anterior.
- Organize o array em ordem crescente.
- Usando JS, insira o conteúdo do array em uma tabela HTML de 9x10.

8	16	35	37	39	40	55	60	102	120
143	144	146	155	161	183	212	217	231	231
235	240	248	262	273	284	290	293	300	307
358	359	385	389	393	396	413	425	434	440
462	466	468	481	491	494	501	503	513	521
530	538	538	543	548	566	569	579	588	628
653	659	680	701	719	731	750	765	799	821
821	832	852	860	876	879	907	915	929	933
933	934	940	945	969	972	976	990	993	997

#Objetos

Arrays, Matrizes e Objetos

Objetos

Objetos no Javascript são parecidos com arrays, mas podem ser indexados por uma string:

```
const pessoa = {
  nome: 'Fulano',
  sobrenome: 'Silva',
  idade: 19,
};
pessoa.salario = 2345.67;

console.log(pessoa.nome);
console.log(pessoa['sobrenome']);

const field = prompt('Informe qual campo deseja saber:');
if (!pessoa[field]){
  alert('Este campo não existe');
}
else {
  alert(`${ field }: ${ pessoa[field] }`);
}
```

Arrays, Matrizes e Objetos

Objetos

Funções úteis para usar com objetos

- Object.keys: Retorna um array com os índices (chaves) do objeto.
- Object.values: Retorna um array com os valores do objeto.
- Object.entries: Retorna um array onde cada element é um array com o par [chave, valor] do objeto.

ATENÇÃO: Objetos não possuem a propriedade **length**.

```
const obj = { foo: 'bar', num: 100 };  
  
console.log(Object.keys(obj)); // [ 'foo', 'num' ]  
console.log(Object.values(obj)); // [ 'bar', 100 ]  
console.log(Object.entries(obj)); // [ ['foo', 'bar'], ['num', 100] ]
```

Arrays, Matrizes e Objetos

Objetos

Podemos percorrer um objeto usando `for...in`

```
const pessoa = {  
  nome: 'Fulano',  
  sobrenome: 'Silva',  
  idade: 19,  
  salario: 2345.67,  
};  
  
let str = '';  
for (let key in pessoa) {  
  const value = pessoa[key];  
  str += `

<span>Chave: ${key}</span>  
    <span>Valor: ${value}</span>  
  </div>`;  
}  
document.body.innerHTML = str;


```

```
div {  
  margin-bottom: 3px;  
  border: 1px solid black;  
  display: flex;  
}  
  
span {  
  width: 200px;  
  padding: 5px 10px;  
}
```

Chave: nome	Valor: Fulano
Chave: sobrenome	Valor: Silva
Chave: idade	Valor: 19
Chave: salario	Valor: 2345.67

#JSON

Arrays, Matrizes e Objetos

Objetos

JSON - Javascript Object Notation

- Formato simples para representação de objetos JS.
- Útil para transferência de informação entre aplicações.
- JSON.stringify: Converte um objeto JS em uma string no formato JSON.
- JSON.parse: Converte uma string no formato JSON em um objeto JS.

```
const pessoa = {  
  nome: 'João',  
  idade: 16,  
  email: 'joao.silva@gmail.com',  
};  
  
const str = JSON.stringify(pessoa);  
console.log(str);  
  
const obj = JSON.parse(str);  
console.log(obj);
```

```
{"nome":"João","idade":16,"email":"joao.silva@gmail.com"}
```

```
▼ {nome: 'João', idade: 16, email: 'joao.silva@gmail.com'} ⓘ  
  email: "joao.silva@gmail.com"  
  idade: 16  
  nome: "João"  
  ► [[Prototype]]: Object
```

Arrays, Matrizes e Objetos

Matrizes

Matrizes é como chamamos arrays multidimensionais.

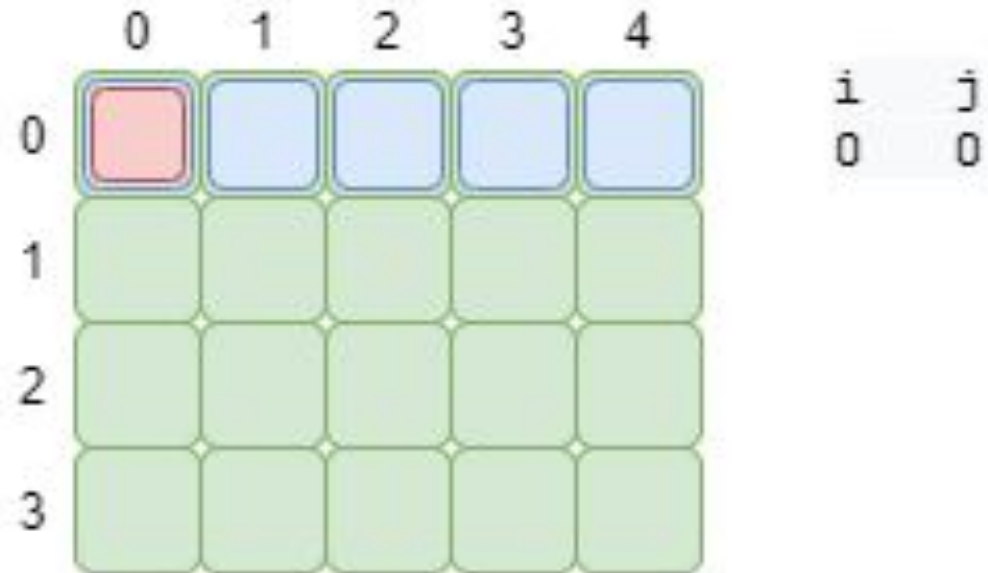
```
const mat = [  
  [ 1, 2, 3, 4, 5],  
  [ 6, 7, 8, 9, 10],  
  [11, 12, 13, 14, 15],  
  [16, 17, 18, 19, 20],  
];
```

Nada mais é que um array, onde cada elemento é um array também.

```
element == row[j] == mat[i][j]
```

Como percorrer?

```
for (let i in mat){  
  const row = mat[i];  
  for (let j in row){  
    const element = row[j];  
    // code  
  }  
}
```



Arrays, Matrizes e Objetos

Matrizes

```
const pessoas = [  
  {  
    nome: 'João',  
    idade: 16,  
    email: 'joao.silva@gmail.com',  
  },  
  {  
    nome: 'Marcelo',  
    idade: 38,  
    email: 'marcelozucca@gmail.com',  
  },  
  {  
    nome: 'Maria',  
    idade: 19,  
    email: 'mama2001@gmail.com',  
  },  
  {  
    nome: 'Rafaela',  
    idade: 12,  
    email: 'rafafanti@gmail.com',  
  },  
  {  
    nome: 'Guilherme',  
    idade: 24,  
    email: 'guilhermecardoso@gmail.com',  
  },  
];
```

É muito comum usar arrays de objetos.

```
const message = document.querySelector('#message');  
document.querySelector('button').addEventListener('click', () => {  
  const input = document.querySelector('input');  
  message.innerHTML = `<span>Pessoa não encontrada</span>`;  
  pessoas.forEach(p => {  
    if (input.value == p.nome){  
      message.innerHTML = `  
        <span>Nome: ${p.nome}</span>  
        <span>Idade: ${p.idade}</span>  
        <span>E-mail: ${p.email}</span>  
      `;  
    }  
  });  
});
```

Arrays, Matrizes e Objetos

Exercício:

- Você irá usar um conteúdo JSON do site [jsonplaceholder](#).
- Escolha um dos conteúdos de exemplo do site: [postagens](#), [comentários](#), [albuns](#), [fotos](#), [tarefas](#) ou [usuários](#).
- Você deverá colocar o JSON dentro de um array.
 - Copie e cole no JS.
- Depois, crie um site que mostre o conteúdo escolhido.
 - Percorra o array, e vá inserindo os elementos na página.
- Dependendo de quais dados foram escolhidos, o layout da página será diferentes, pois os dados não são os mesmos.
 - Ex: As fotos contém links para as imagens e títulos, enquanto os usuários contém endereços e emails.

```
[
  {
    "id": 1,
    "name": "Leanne Graham",
    "username": "Bret",
    "email": "Sincere@april.biz",
    "address": {
      "street": "Kulas Light",
      "suite": "Apt. 556",
      "city": "Gwenborough",
      "zipcode": "92998-3874",
      "geo": {
        "lat": "-37.3159",
        "lng": "81.1496"
      }
    },
    "phone": "1-770-736-8031 x56442",
    "website": "hildegard.org",
    "company": {
      "name": "Romaguera-Crona",
      "catchPhrase": "Multi-layered client-server neural-net",
      "bs": "harness real-time e-markets"
    }
  },
  {
    "id": 2,
    "name": "Ervin Howell",
    "username": "Antonette",
    "email": "Shanna@melissa.tv",
    "address": {
      "street": "Victor Plains",
      "suite": "Suite 879",
      "city": "Wisokyburgh",
      "zipcode": "90566-7771",
      "geo": {
        "lat": "-43.9509",
        "lng": "-34.4618"
      }
    },
    "phone": "010-692-6593 x09125",
    "website": "anastasia.net",
    "company": {
      "name": "Deckow-Crist",
      "catchPhrase": "Proactive didactic contingency",
      "bs": "synergize scalable supply-chains"
    }
  },
  {
    "id": 3,
    "name": "Clementine Bauch",
    "username": "Samantha",
    "email": "Nathan@yesenia.net",
    "address": {
      "street": "Douglas Extension",
      "suite": "Suite 847",
      "city": "McKenziehaven",

```

Professor Fábio Luís da Silva Santos

www.ifsul.edu.br
fabiosantos@ifsul.edu.br