

Programação II

Vetores e Objetos em JavaScript

Prof. Fernando Mattioli



ECOSSISTEMA
BRASÍLIA
EDUCACIONAL

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Vetores

- ▶ Vetores são utilizados para armazenar múltiplos valores em uma única variável.
- ▶ Em JavaScript, vetores são indexados por números, iniciando em 0.
- ▶ Em JavaScript, vetores podem armazenar conteúdos de diferentes tipos.
- ▶ Exemplos: `https://www.w3schools.com/js/js_arrays.asp`.

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Exemplos

Inicialização

- Vetores são inicializados utilizando colchetes ou a palavra reservada *new*:

```
// Inicialização com colchetes
```

```
var cores = ["Verde", "Branco", "Vermelho"];
```

```
// Inicialização com new
```

```
var cores = new Array("Verde", "Branco", "Vermelho");
```

Inicialização

- ▶ Vetores são inicializados utilizando colchetes ou a palavra reservada *new*:

```
// Declaração do vetor vazio  
var cores = []  
cores[0] = "Verde";  
cores[1] = "Branco";  
cores[2] = "Vermelho";
```

Acessando elementos

- ▶ Elementos são acessados por seu índice numérico (iniciado em 0):

```
// Inicialização com colchetes  
var cores = ["Verde", "Branco", "Vermelho"];  
  
console.log(cores[0]); // Primeiro elemento  
console.log(cores[1]); // Segundo elemento  
console.log(cores[2]); // Terceiro elemento
```


Métodos/propriedades úteis

- ▶ *length*: retorna o tamanho do vetor.
- ▶ *push(x)*: adiciona o elemento x ao vetor.
- ▶ *sort()*: ordena os elementos do vetor.

Métodos/propriedades úteis

```
var vet = [5, 3, 2];  
  
vet.push(4);  
  
vet.sort();  
  
for (var i = 0; i < vet.length; ++i) {  
    console.log(vet[i]);  
}
```

Métodos/propriedades úteis

- ▶ *pop()*: remove e retorna o último elemento do vetor.
- ▶ *shift()*: remove e retorna o primeiro elemento do vetor.
- ▶ *unshift(x)*: insere o elemento x no início do vetor.

Métodos/propriedades úteis

```
var vet = [5, 3, 2];  
  
console.log(vet.pop());  
console.log(vet.shift());  
  
vet.unshift(1);  
  
for (var i = 0; i < vet.length; ++i) {  
    console.log(vet[i]);  
}
```

Métodos/propriedades úteis

- ▶ *slice(a, b)*: retorna parte do vetor, começando no índice *a* e terminando no índice *b - 1*.
- ▶ *splice()*: permite adicionar ou remover elementos ao vetor.
- ▶ *concat(v)*: permite concatenar o vetor *v* ao vetor atual.

Métodos/propriedades úteis

```
var v1 = [5, 4, 3, 2, 1];  
var v2 = v1.slice(1,3);  
for (var i = 0; i < v2.length; ++i)  
    console.log(v2[i]);  
  
v1.splice(2, 0, 5); // Adiciona o valor 5, na posição 2  
for (var i = 0; i < v1.length; ++i)  
    console.log(v1[i]);
```

Métodos/propriedades úteis

```
v1.splice(2, 1); // Remove 1 elemento, na posição 2  
for (var i = 0; i < v1.length; ++i)  
    console.log(v1[i]);  
  
var v3 = v1.concat(v2);  
for (var i = 0; i < v3.length; ++i)  
    console.log(v3[i]);
```

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Objetos

- ▶ Objetos são representações de elementos do mundo real.
- ▶ Exemplo: um carro é um objeto.

Objetos

- ▶ Um carro possui atributos (propriedades), tais como peso e cor, além de comportamento (métodos), tais como acelerar e parar.
- ▶ Todos os carros possuem as mesmas propriedades, porém os valores das propriedades diferem de carro para carro.

Objetos

- ▶ Todos os carros possuem os mesmos métodos, porém os métodos são executados em momentos diferentes.
- ▶ Objetos são declarados como variáveis. No entanto, podem receber muitos valores simultaneamente.
- ▶ Em JavaScript, declara-se objetos utilizando chaves, separando as propriedades dos valores por “:”.

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Inicialização

```
// Declara um carro da marca Fiat, modelo Palio e cor branca.  
var carro = {marca:"Fiat", modelo:"Palio", cor:"Branco"};  
  
// Declara uma pessoa com nome João, sobrenome Silva e idade 20  
anos.  
var pessoa = {nome:"João", sobrenome:"Silva", idade:20};
```

Acesso às propriedades

- As propriedades de um objeto podem ser acessadas utilizando-se o ponto (“.”) ou os colchetes (“[]”).

```
// Declara uma pessoa com nome João, sobrenome Silva e idade 20.  
var pessoa = {nome:"João", sobrenome:"Silva", idade:20};  
  
console.log("Nome: " + pessoa.nome);  
console.log("Idade: " + pessoa["idade"]);
```

Métodos

- ▶ Métodos podem ser declarados na inicialização dos objetos.

```
var pessoa = {  
  nome: "João",  
  sobrenome: "Silva",  
  idade: 20,  
  nomeCompleto: function() {  
    return this.nome + " " + this.sobrenome;  
  }  
};
```

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Vetores e objetos

- ▶ É possível criar vetores de objetos:

```
var p1 = {nome:"João", sobrenome:"Silva", idade:20};  
var p2 = {nome:"Maria", sobrenome:"Silva", idade:30};  
var p3 = {nome:"José", sobrenome:"Silva", idade:40};  
var v = [p1, p2, p3];  
for (var i = 0; i < v.length; ++i)  
    console.log(v[i].nome + " " + v[i].sobrenome + " tem " +  
        v[i].idade + " anos.");
```

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Exercício

1. Resolver os 5 exercícios disponíveis em

`https://www.w3schools.com/js/js_arrays.asp`.

Exercício

2. Resolver os 3 exercícios disponíveis em

`https://www.w3schools.com/js/js_objects.asp`.

Sumário

Vetores

Exemplos

Objetos

Exemplos

Vetores e objetos

Exercícios

Referências

Referências

- ▶ Deitel, P.; Deitel, H. *Ajax, Rich Internet Applications e desenvolvimento Web para programadores*. São Paulo, Pearson Prentice Hall, 2008.
- ▶ https://www.w3schools.com/js/js_objects.asp
- ▶ https://www.w3schools.com/js/js_arrays.asp