

Objetos

INTRODUÇÃO A CRIAÇÃO E USO DE OBJETOS EM JAVASCRIPT

AUTOR(A): PROF. DANIEL FERREIRA DE BARROS JUNIOR

Objetos

A ideia de criar funções permite ir além na programação em JavaScript. Entre estes passos vamos verificar o uso de objetos. A finalidade de criar objetos é geralmente representar coisas do mundo real, os objetos. Podemos representar certas generalizações, abstratas, mais não serão tratadas neste momento.

Imagine um automóvel. Ele é um veículo de transporte terrestre. Pois bem, ele possui diversas características ou propriedades e pode executar diversas ações, ou métodos.

Vamos analisar o código a seguir:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>Tópico 7</title>
5.         <meta charset="UTF-8">
6.     </head>
7.     <body>
8.         <div>Programação de Interfaces (aula 10)</div>
9.         <script type="text/javascript">
10.             var carro = {
11.                 nome: "Camaro",
12.                 marca: "GM",
13.                 potencia: 300,
14.                 cor: "Amarelo",
15.                 preco: 200000
16.             };
17.         </script>
18.     </body>
19. </html>
```

Atributos

Neste exemplo, um objeto com o nome de carro é criado, com determinadas características ou propriedades.

Estas propriedades podem ser acessadas facilmente utilizando-se a forma:

```
objeto.propriedade;
```

Assim como nas linguagens orientadas a objetos, estes objetos possuem certos atributos, ou propriedades.

Imagine um objeto carro, de cor branca. Esta cor é uma característica dos carros, que em objetos a cor é um atributo (ou propriedade) do objeto carro.

Vamos analisar a continuidade deste script, no código a seguir:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>Tópico 7</title>
5.         <meta charset="UTF-8">
6.     </head>
7.     <body>
8.         <div>Programação de Interfaces (aula 10)</div>
9.         <script type="text/javascript">
10.             var carro = {
11.                 modelo: "Camarom",
12.                 marca: "GG",
13.                 potencia: 300,
14.                 cor: "Amarelo",
15.                 preco: 200000
16.             };
17.
18.             document.write("<br>Comprei um " + carro.modelo + ", da cor " +
19.                 document.write("<br>Já tive outros veículos da " + carro.marca
20.         </script>
21.     </body>
22. </html>
```

O resultado para este script é apresentado na imagem a seguir:

Programação de Interfaces (aula 10)

Comprei um Camarom, da cor Amarelo!

Já tive outros veículos da GG. investi mais de R\$ 200000!

Podemos criar objetos para nos auxiliar na programação em JavaScript. E aos objetos podemos ainda criar ações os métodos. Vamos analisar o código estendido:

```

1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>Tópico 7</title>
5.         <meta charset="UTF-8">
6.     </head>
7.     <body>
8.         <div>Programação de Interfaces (aula 10)</div>
9.         <script type="text/javascript">
10.             var carro = {
11.                 modelo: "Camarom",
12.                 marca: "GG",
13.                 potencia: 300,
14.                 cor: "Amarelo",
15.                 preco: 200000,
16.                 modeloMarca: function(){
17.                     return this.modelo + " da marca " + this.marca;
18.                 }
19.             };
20.
21.             document.write("<br>Comprei um " + carro.modelo + ", da cor " +
22.                 document.write("<br>Já tive outros veículos da " + carro.marca
23.                 document.write("<br>Todos querem ver o " + carro.modeloMarca()
24.         </script>
25.     </body>
26. </html>

```

Para este script temos o seguinte resultado:

Programação de Interfaces (aula 10)

Comprei um Camarom, da cor Amarelo!
Já tive outros veículos da GG. Investi mais de R\$ 200000!
Todos querem ver o Camarom da marca GG!

Observe que criamos um método modeloMarca (linha 16), onde retornamos algumas propriedades do objeto carro, como o modelo e a marca. Verifique também que utilizamos o objeto this.

this.propriedade;

O comando `this` é uma palavra chave que representa o próprio objeto e posteriormente uma propriedade.

Métodos

Os métodos são ações executadas por objetos.

Considerando um carro como um objeto, algumas ações possíveis seriam:

- `andar()`
- `parar()`
- `virar_Direita()`
- `virar_Esquerda()`
- `estacionar()`

São métodos (ações) do objeto carro.

A seus objetos instanciados, sempre que necessário, é possível criar métodos que correspondam a uma ação do objeto.

Veja mais sobre objetos, atributos e métodos em:

https://www.w3schools.com/js/js_objects.asp (https://www.w3schools.com/js/js_objects.asp)

DICA:

Veja mais sobre objetos, atributos e métodos em:

https://www.w3schools.com/js/js_objects.asp (https://www.w3schools.com/js/js_objects.asp)

https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects

(https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects)

Conclusão

A criação e uso de objetos facilita e organiza a criação de scripts em JavaScript. Apresentamos a criação de propriedades e métodos de um objeto. A sua utilização é bem intuitiva e simples. Crie os seus objetos para praticar!

ATIVIDADE

Escolha a alternativa que obtenha algum resultado correto ao script a seguir:

```
<script type="text/javascript">
```

```
    var carro = {  
        nome: "Camaro",  
        marca: "GM",  
        potencia: 300,  
        cor: "Amarelo",  
        preco: 200000  
    };
```

```
</script>
```

- A. carro.completo();
- B. carro.valor;
- C. carro.marca;
- D. carro,marca();

ATIVIDADE

Qual a finalidade da utilização do comando "this.marca"?

- A. alterar a marca do carro
- B. acessar somente a marca, de todos os carros.
- C. acessar a propriedade marca do objeto em questão
- D. criar um novo objeto com a propriedade marca.

ATIVIDADE

Considerando objetos instanciados, selecione a opção que melhor exemplifica o significado de atributos.

- A. Atributo corresponde as características de um objeto.

- B. Atributo corresponde aos métodos de um objeto.
- C. Atributo corresponde as características de uma classe.
- D. Atributo corresponde aos métodos de um classe.

REFERÊNCIA

- MORRISON, M. Use a cabeça JavaScript. 5º Ed. Rio de Janeiro: Alta Books, 2012. 606 p.
- OLIVIERO. C. A. J. Faça um site JavaScript orientado por projeto. 6º ed. São Paulo: Érica, 2010. 266 p.
- ZAKAS, Nicholas C. JavaScript de alto desempenho. 8º Ed. São Paulo: Novatec, 2010. 245 p.

