

Objetos

Um objeto pode ser utilizado para guardar várias informações referente a um único elemento.

Um objeto pode possuir propriedade e métodos.

Imagine que em um sistema tenha diferentes informações a cerca de um produto, nome, preço e quantidade, a primeiro momento você pode estar pensando em armazenar essas informações em uma variável, mas não seria a melhor forma. Poderíamos criar uma um objeto chamado produto que conteria todas essas informações. Um objeto é criado da seguinte forma:

```
var produto1 = new Object();

produto1.nome = 'Caneta';
produto1.preco = 1.25;
produto1.quantidadeEstoque = 25;
```

A forma utilizada a cima para criar um objeto é a maneira formal.

Quando definimos um objeto e em seguida colocamos um ponto e alguma coisa, denominamos que é uma propriedade do objeto. Uma propriedade pode ser utilizada como uma “variável”, podendo armazenar informações;

Problema

Mostrar na tela todos os dados de um produto específico, contendo quantidade em estoque e preço. Estes dados estarão agrupados em um objeto.

```
var produto1 = new Object();
var $tela = document.querySelector('.tela');

produto1.nome = 'Banana';
produto1.preco = 3.5;
produto1.quantidade = 20;
```

```

mostrarProduto(produto1);

function mostrarProduto(produto){
    var exibir = null;

    exibir = produto.nome + '<br>';
    exibir += produto.preco + '<br>';
    exibir += produto.quantidade + '<br>';

    $tela.innerHTML = exibir;
}

```

Métodos

Um método é basicamente uma função que pertence a um objeto. Para definir uma função em um objeto pode-se fazer da seguinte forma:

```

var carro = new Object();

carro.ligar = function(){
    //Linhas de código;
}

//Invocando essa função
carro.ligar();

```

Um outro exemplo seria relacionado ao exemplo anterior do sistema de produtos, imagine que ao comprar um produto ele tenha que diminuir a quantidade no estoque.

```

var produto = new Object();

produto.nome = 'Café';
produto.quantidade = 20;
produto.comprar = function(qtd){
    if(!qtd){
        qtd = 1;
    }
    this.quantidade = this.quantidade - qtd;
}

```

Você deve está se perguntando o que significa `this`. This a grosso modo nada mais é do que uma referencia ao próprio objeto.

Exercício 01

Mostrar na tela os dados de produtos, contendo a quantidade em estoque e o preço. Esses dados estarão agrupados em um array de objetos.

```
var produtos = [  
  {'nome': 'Maça', 'valor': 3.80, 'quantidade': 20},  
  {'nome': 'Kiwi', 'valor': 4.50, 'quantidade': 15},  
  {'nome': 'Péra', 'valor': 3.20, 'quantidade': 10},  
  {'nome': 'Melancia', 'valor': 2.40, 'quantidade': 50}  
];  
  
var $resultado = document.querySelector('.resultado');  
var msg = ''  
  
for(var i = 0; i < produtos.length; i++){  
  msg += 'Produto: ' + produtos[i].nome + '<br>';  
  msg += 'Valor: ' + produtos[i].valor + '<br>';  
  msg += 'Estoque: ' + produtos[i].quantidade + '<br>';  
  msg += '<br>'  
}  
  
$resultado.innerHTML = msg;
```

Exercício 02

criar um array com três objetos. Cada objeto deverá ter duas propriedades (nome e email). Depois, basta mostrar na tela.

```
var $saida = document.querySelector('.saida');  
  
var pessoas = [  
  {'nome': 'Vinicius Sousa', 'email': 'viniciuszika@gmail.com'},  
  {'nome': 'Gordofredo das banhas', 'email': 'gordinho@bol.com'},  
  {'nome': 'Jubileia', 'email': 'jubi@leia.com.br'}  
];  
  
var texto = ''  
  
for(var i = 0; i < pessoas.length; i++){  
  texto += 'Nome: ' + pessoas[i].nome + '<br>';  
  texto += 'E-mail: ' + pessoas[i].email + '<br>';  
  texto += '<br>';  
}  
  
$saida.innerHTML = texto;
```

