

Métodos úteis

Ajax & JSON

Bem-vindo ao jQuery Festas

BEBIDA EM DESTAQUE

Destaque

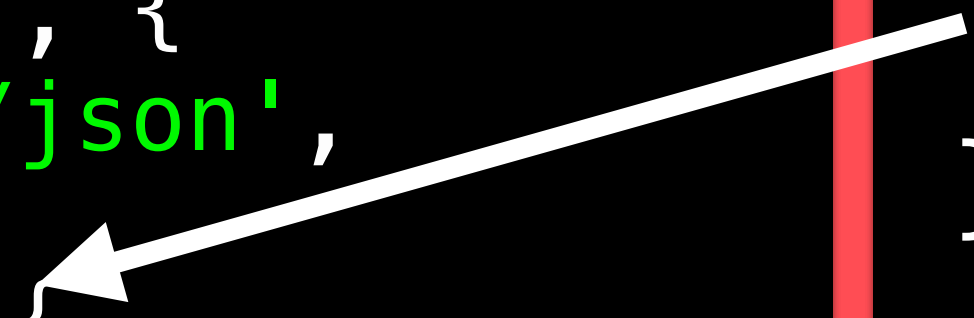
```
<h1>Bem-vindo ao jQuery Festas</h1>
<button>Bebida em Destaque</button>

<div class="destaque">
  <h2>Destaque</h2>
  <img src="" />
  <p></p>
</div>
```

Ajax & JSON

```
$( 'button' ).on( 'click', function () {  
    $.ajax( 'bebidas/favoritas/1', {  
        contentType: 'application/json',  
        dataType: 'json',  
        success: function( result ) {  
            var destaque = $( '.destaque' );  
            destaque.find( 'p' ).html( result.nome );  
            destaque.find( 'img' )  
                .attr( 'src', result.imagem );  
        }  
    } );  
});
```

```
{  
    "nome": "Lemon Breeze",  
    "imagem": "imagens/yellow.gif"  
}
```



*e se existíssem
múltiplos destaques?*

```
<h1>Bem-vindo ao jQuery Festas</h1>  
<button>Bebida em Destaque</button>  
  
<div class="destaque">  
    <h2>Destaque</h2>  
    <img src="" />  
    <p></p>  
</div>
```



Drinks

Bem-vindo ao jQuery Festas

BEBIDA EM DESTAQUE

Destaque

Destaque

Destaque

JSON com vários resultados

application.js

```
[  
  {  
    "nome": "Lemon Breeze",  
    "imagem": "imagens/yellow.gif"  
  },  
  {  
    "nome": "Dark Juice",  
    "imagem": "imagens/black.gif"  
  },  
  {  
    "nome": "Chai Tea",  
    "imagem": "imagens/chai.gif"  
  }  
]
```

result

```
success: function(result) {  
  var destaque = $('#destaque');  
  destaque.find('p').html(result.nome);  
  destaque.find('img')  
    .attr('src', result.imagem);  
}
```

*precisa-se de um laço
para cada destaque*

*O resultado é um
array de objetos*

\$.each -> iteração sobre um array

application.js

```
success: function(result) {  
  $.each(result, function(indice, bebida) {  
    var destaque = $('<div>.destaque-</div>' + indice);  
    destaque.find('<div>p</div>').html(bebida.nome);  
    destaque.find('<div>img</div>')  
      .attr('src', bebida.imagem);  
  })  
}
```

index.html

```
<li class="destaque-0">  
  ...  
</li>  
  
<li class="destaque-1">  
  ...  
</li>  
  
<li class="destaque-2">  
  ...  
</li>
```

Executa a função para cada objeto da coleção

```
$.each(colecao, function(<indice>, <objeto>) {});
```

**E quando a quantidade de
informação for variável?**

E quando a quantidade de informação for variável?

jQuery Festas: versão administrativa

ATUALIZAR PEDIDOS

Pedido: 45

Entregue

Pedido: 47

Em preparação

Transforme um array de objetos em HTML

application.js

```
$( '.atualizar-pedidos' ).on( 'click', function() {  
    $.ajax( 'status/todos.json', {  
        contentType: 'application/json',  
        dataType: 'json',  
        success: function (result) {...}  
    });  
});
```

*o resultado será um
array de objetos*

```
$.getJSON(url, success);
```

```
$.getJSON( 'status/todos.json', function(result) {});
```

```
[  
    {  
        "numero": "50",  
        "status": "Não iniciado"  
    },  
    {  
        "numero": "45",  
        "status": "Entregue"  
    },  
    {  
        "numero": "47",  
        "status": "Cancelado"  
    }  
]
```

result

Transforme um array de objetos em HTML

application.js

```
$( '.atualizar-pedidos' ).on( 'click', function() {  
    $.getJSON( 'status/todos.json', function( result ) {  
        var pedidos = ???  
        $( 'ul' ).html( pedidos );  
    });  
});
```

```
<li>  
    <h3>numero</h3>  
    <p>status</p>  
</li>  
<li>  
    <h3>...</h3>  
    <p>...</p>  
</li>  
<li>  
    <h3>...</h3>  
    <p>...</p>  
</li>
```

é preciso transformar
para um formato HTML

```
[  
    {  
        "numero": "50",  
        "status": "Não iniciado"  
    },  
    {  
        "numero": "45",  
        "status": "Entregue"  
    },  
    {  
        "numero": "47",  
        "status": "Cancelado"  
    }  
]
```

result

Array com \$.map()

a coleção pode ser um array

```
$.map(colecao, function(<item>, <indice>) {});
```

a ordem é diferente do each

o método map um array modificado a partir do array passado com o parâmetro

```
var numeros = [1,2,3,4]
```

```
var numeros-novos = $.map(numeros, function(item, indice) { return item + 1 });
```

numeros → [1, 2, 3, 4]

numeros-novos → [2, 3, 4, 5]

Do JSON para o HTML

```
{  
  "numero": "50",  
  "status": "Não iniciado"  
},
```

```
{  
  "numero": "45",  
  "status": "Entregue"  
},
```

```
{  
  "numero": "47",  
  "status": "Cancelado"  
}
```

```
$.map(result, function(pedido, indice) {  
  var item-pedido = $("<li></li>");  
  return item-pedido;  
});
```


Do JSON para o HTML

```
{  
  "numero": "50",  
  "status": "Não iniciado"  
},
```

```
{  
  "numero": "45",  
  "status": "Entregue"  
},
```

```
{  
  "numero": "47",  
  "status": "Cancelado"  
}
```

```
$.map(result, function(pedido, indice) {  
  var item-pedido = $("<li></li>");  
  $('<h3> Pedido: ' + pedido.numero + '</h3>').appendTo(item-pedido);  
  $('<p>' + pedido.status + '</p>').appendTo(item-pedido);  
  return item-pedido;  
});
```


```
<li>  
  <h3>Pedido: 50</h3>  
  <p>Não iniciado</p>  
</li>
```

```
<li>  
  <h3>Pedido: 45</h3>  
  <p>Entregue</p>  
</li>
```

```
<li>  
  <h3>Pedido: 47</h3>  
  <p>Cancelado</p>  
</li>
```

Do JSON para o HTML

```
$( '.atualizar-pedidos' ).on( 'click', function () {  
    $.getJSON( 'status/todos.json', function (result) {  
        var pedidos = $.map(result, function (pedido, indice) {  
            var item_de_pedidos = $( '<li></li>' );  
            $( '<h2>Pedido: ' + pedido.numero + '</h2>' ).appendTo( item_de_pedidos );  
            $( '<p>' + pedido.status + '</p>' ).appendTo( item_de_pedidos );  
            return item_de_pedidos;  
        });  
        $( 'ul' ).html( pedidos );  
    });  
});
```



retorna-se a lista
alterada

jQuery Festas: versão administrativa

ATUALIZAR PEDIDOS

Pedido: 45

Entregue

Pedido: 47

Em preparação



Exercícios 7 e 8