

Ajax Básico

Revelar conteúdo

Drink da Noite

Lemon Breeze

R\$ 15.00

VEJA A RECEITA

Código sem Ajax

index.html

```
<li class="bebida promocao" data-preco="15.00">
  <h3>Lemon Breeze</h3>
  <p>R$ 15.00</p>
  <button>Veja a Receita</button>
  <div class="receita">
    <p>Um drinks mais que ... </p>
    <a href="#" class="expandir">Veja a bebida</a>
    
  </div>
</li>
```

Qual seria a forma
de apresentar este conteúdo
de maneira dinâmica?

apresenta o elemento oculto
quando é clicado

application.js

```
$('.bebidas').on('click', 'button', function () {
  $(this).closest(".bebida").find(".receita").slideDown();
});
$('.bebidas').on('click', '.expandir', function() {
  $(this).closest(".bebida").find("img").show();
});
```

exibe a imagem
quando é clicado

\$.ajax(url [, settings])

executa uma requisição

HTTP assíncrona

url

obrigatória

settings

configurações opcionais

conjunto chave/valor que configura a requisição AJAX

- **success:** só executa quando a chamada a url funciona corretamente

Código sem Ajax

index.html

```
<li class="bebida promocao" data-preco="15.00">  
  <h3>Lemon Breeze</h3>  
  <p>R$ 15.00</p>  
  <button>Veja a Receita</button>  
  <div class="receita">  
  
  </div>  
</li>
```

\$.ajax(url [, settings])

Utilizando uma requisição AJAX application.js

```
$('.beidas').on('click', 'button', function () {  
    $.ajax('receita.html', {  
        success: function(response) {  
            $('.receita').html(response).slideDown();  
        }  
    });  
});
```

só executa quando
a resposta retorna
com sucesso

receita.html

```
<p>Um drinks ...</p>  
<a href="#" class="expandir">Veja a bebida</a>  

```

resposta, que não
é uma página HTML
completa

\$.get(url, success)

executa uma requisição

HTTP assíncrona via HTTP GET

url

obrigatória

success

função a ser executada
quando a resposta for
retornada

Atalho com \$.get

```
$( '.beidas' ).on( 'click', 'button', function () {  
    $.ajax( 'receita.html', {  
        success: function( response ) {  
            $( '.receita' ).html( response ).slideDown();  
        }  
    } );  
});
```

equivalentes

```
$( '.beidas' ).on( 'click', 'button', function () {  
    $.get( 'receita.html', function( response ) {  
        $( '.receita' ).html( response ).slideDown();  
    } );  
});
```


Como enviar parâmetros?

sempre recupera o mesmo arquivo

```
$.ajax('receita.html', {  
    success: function(response) {  
        $(".receita").html(response).slideDown();  
    }  
});
```

Como criar um site em que a requisição levasse em conta o ID da receita?

```
receita.html?id=123
```

Como enviar parâmetros?

sempre recupera o mesmo arquivo

```
$.ajax('receita.html?id=123', {  
  success: function(response) {  
    $(".receita").html(response).slideDown();  
  }  
});
```

```
$.ajax('receita.html', {  
  success: function(response) {  
    $(".receita").html(response).slideDown();  
  },  
  data: {"id": 123}  
});
```

Como enviar parâmetros?

```
<div class="receita" data-id="123">
```

```
$.ajax('receita.html', {  
    success: function(response) {  
        $(".receita").html(response).slideDown();  
    },  
    data: {id: $(".receita").data("id")}  
});
```

\$.ajax(url [, settings])

*executa uma requisição
HTTP assíncrona*

- **success:** só executa quando a chamada a url funciona corretamente
- **error:** executa quando ocorre um timeout, erro no servidor, ou a requisição é abortada
- **timeout:** indica em ms quanto a requisição deve esperar
- **beforeSend:** executa antes da requisição ajax
- **complete:** executa após a execução da requisição com sucesso ou erro

Manipulando requisições com falha

```
$( '.beidas' ).on( 'click', 'button', function () {  
    $.ajax( 'http://www.fa7.edu.br/pos-si/receita.html', {  
        success: function( response ) {  
            $( ".receita" ).html( response ).slideDown();  
        },  
        error: function( request, errorType, errorMessage ) {  
            alert( 'Error: ' + errorType + ', mensagem: ' + errorMessage );  
        }  
    });  
});
```

- *error: executa quando ocorre um timeout, erro no servidor, ou a requisição é abortada*

Manipulando requisições com falha

```
$( '.bebidas' ).on( 'click', 'button', function () {  
    $.ajax( 'http://www.fa7.edu.br/pos-si/receita.html', {  
        success: function( response ) {  
            $( ".receita" ).html( response ).slideDown();  
        },  
        error: function( request, errorType, errorMessage ) { ... },  
        timeout: 3000,  
        beforeSend: function() {  
            $( ".bebidas" ).addClass( "carregando" );  
        }  
        complete: function() {  
            $( ".bebidas" ).removeClass( "carregando" );  
        }  
    });  
});
```

Exercícios 1 e 2