

CURSO AJAX

SYS4SOFT (João Ribeiro)

https://www.youtube.com/watch?v=cQoGow-UwDc&list=PLXik_5Br-zO9hDBdGmzX5XJ9LT6lcHPiG

Resumo do curso feito por Roberto Pinheiro

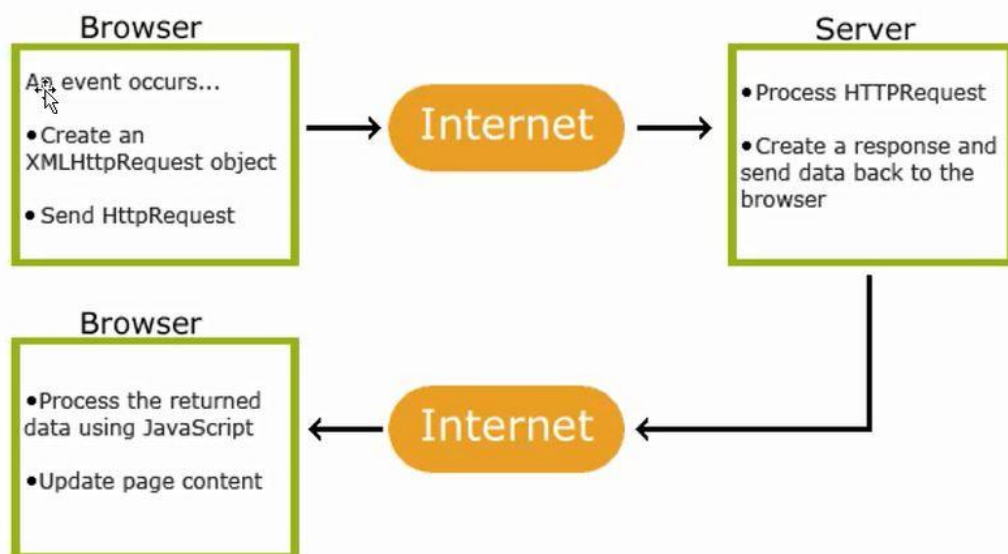
Aula 01 - Introdução

Uma das técnicas essenciais no mundo JavaScript é o AJAX. Com ele podemos criar sites mais eficientes e agradáveis para o usuário, carregando apenas trechos específicos da página, em lugar de carregá-la por completo.

AJAX é acrônimo em língua inglesa de "*Asynchronous Javascript and XML*", que em português significa "**Javascript e XML Assíncronos**". Designa um conjunto de técnicas para programação e desenvolvimento web que utiliza tecnologias como Javascript e XML para carregar informações de forma assíncrona.

Asynchronous JavaScript e XML;

- Não é uma linguagem de programação;
- Técnica de programação em JavaScript;
- Receber e enviar dados para o servidor de forma assíncrona;
- Através do DOM, atualiza a página web sem ter que recarregar.



Aula 02 – Exemplo simples

Arquivo **aula02.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 02</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula02.css">
  </head>
  <body>
    <div id="container">
      <h1>Aula 02</h1>
      <h2>Exemplo simples</h2>
      <div id="caixa">Este é o texto da caixa.</div>
      <button id="btn">Clicar</button>
    </div>
    <script type="text/javascript" src="js/aula02.js"></script>
  </body>
</html>
```

Arquivo **aula02.css**

```
#caixa{
  margin: 20px;
}
#btn {
  padding: 15px;
  font-family: Verdana;
  font-size: 14px;
  margin: 10px 20px;
}
```

Arquivo **dados.txt**

Esta é a frase do servidor.

Arquivo **aula02.js**

```
window.onload = function () {  
  
    document.getElementById('btn').addEventListener('click', btn_click);  
  
};  
  
function btn_click() {  
  
    // document.getElementById("caixa").innerHTML = "Este é o texto alterado";  
  
    /* ajax */  
  
    /* criamos o objeto Request */  
    let pedido = new XMLHttpRequest();  
  
    /* definir a função que vai atuar mediante a resposta */  
    pedido.onreadystatechange = function () {  
        document.getElementById('caixa').innerHTML = this.responseText;  
    }  
  
    pedido.open("GET", "dados/dados.txt", true);  
    pedido.send();  
  
}
```

Para rodar: http://localhost/curso_ajax1/aula02.html



Clicando no botão:

Aula 02

Exemplo simples

Esta é a frase do servidor.

Clicar

Aula 03 - O objeto XMLHttpRequest

Todos os browsers modernos suportam o objeto XMLHttpRequest.

O objeto XMLHttpRequest possui propriedades (atributos) e métodos e pode ser usado para trocar dados com um servidor. Isso significa que é possível atualizar partes de uma página da Web, sem recarregar a página inteira.

Por motivos de segurança, a tecnologia Ajax não permite que dentro do seu servidor seja feito uma solicitação (request) para um outro domínio (outro servidor).

Versões antigas do Internet Explorer (IE5 and IE6)

Versões antigas do Internet Explorer (IE5 e IE6) usam um objeto ActiveX em vez do objeto XMLHttpRequest:

```
variavel = new ActiveXObject("Microsoft.XMLHTTP");
```

Verifique se o navegador suporta o objeto XMLHttpRequest e caso utilize os navegadores IE5 e o IE6 crie um objeto ActiveX:

Exemplo:

```
if (window.XMLHttpRequest) {  
    // code for modern browsers  
    xmlhttp = new XMLHttpRequest();  
} else {  
    // code for old IE browsers  
    xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");  
}
```

Métodos do objeto XMLHttpRequest

new XMLHttpRequest(): Cria um objeto XMLHttpRequest.

open(method,url,async,user,psw): Especifica a requisição (pedido).

- **method:** tipo da requisição (GET ou POST).
- **url:** o local do arquivo no servidor (arquivo de dados → texto ou script PHP para, por exemplo, fazer acesso a uma base dados).
- **async:** true (assíncrono) ou false (síncrono)
- **user:** nome do usuário (opcional)
- **psw:** senha (opcional)

send(): Envia o pedido para o servidor como requisição do tipo GET.

send(string): Envia o pedido para o servidor como uma requisição do tipo POST.

Propriedades do objeto XMLHttpRequest

onreadystatechange: Define uma ação (chama uma função) quando essa propriedade é alterada

readyState: Significa a etapa em que está o nosso pedido.

- 0: requisição não inicializada
- 1: conexão estabelecida com o servidor
- 2: o servidor recebeu a requisição
- 3: o servidor está processando a requisição
- 4: requisição finalizada e a resposta está pronta

status: Retorna um número que identifica o status da requisição.

- 200: "OK" (caso a resposta chegou em condições)
- 403: "Forbidden" (caso a resposta não chegou em condições)
- 404: "Not Found" (caso a resposta sequer foi encontrada)

responseText: Retorna o dado como uma string.

responseXML: Retorna o dado como um XML.

statusText: Retorna o status no formato texto (OK, Forbidden ou Not Found).

Aula 04 – Acrescentando controle no pedido

Enviar o pedido para o servidor

Para enviar um pedido para o servidor, usamos os métodos `open()` e `send()` do objeto `XMLHttpRequest`:

```
pedido.open("GET", "dados/dados_lorem.txt", true);
pedido.send();
```

GET ou POST?

GET é mais simples e rápido do que POST, e pode ser usado na maioria dos casos.

Entretanto, sempre use requisição POST:

- Quando o texto for uma instrução e não um arquivo (por exemplo: instrução para atualizar um arquivo ou banco de dados no servidor).
- para enviar uma grande quantidade de dados para o servidor (POST não tem limitação no tamanho).
- para enviar dados de um input (que pode conter caracteres desconhecidos), POST é mais robusto e seguro do que o GET.

Arquivo **aula04.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 04</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula04.css">
  </head>
  <body>
    <div id="container">
      <h1>Aula 04</h1>
      <h2>Acrescentando controle no pedido</h2>
      <button id="btn" class="espaco">Clicar</button>
      <div id="caixa" class="espaco">Este é o texto da caixa.</div>
    </div>
    <script type="text/javascript" src="js/aula04.js"></script>
  </body>
</html>
```

Arquivo **aula04.css**

```
#caixa{  
    margin: 20px;  
}  
  
#btn {  
    padding: 15px;  
    font-family: Verdana;  
    font-size: 14px;  
    margin: 10px 20px;  
}
```

Arquivo **dados_lorem.txt**

Esta é a frase do servidor.

Lorem ipsum dolor sit amet, has et soluta commodo, suas alienum omittam pri ad, an errem euripidis argumentum sed. Sea no modo prima, illud postulant no vix. Cu molestie volutpat persequeris mei. Labores vocibus appetere cum an. Cum epicuri consetetur te, an ipsum nulla alterum pri.

In sea dolore doming, scripta numquam indoctum in sea. Paulo laoreet vel at, in duis mentitum liberavisse vis. Exerci impetus aliquando est no, fierent mandamus philosophia quo an. Eam quem sapientem id. Pri an aequae facilisis scriptorem, qui stet dolores tractatos ex. Sed eu latine scripta voluptaria, his ex vide sale, ne eam ipsum doming.

Vis ea debet omittam splendide. Nec vide apeirian cu. Molestie vulputate persecuti quo eu, eum cibo senserit in, sea alii phaedrum scriptorem ei. Id duis nostrud nam, cu vix mentitum praesent abhorreant. Mel periculis inciderint appellantur et, ea eos recusabo prodesset.

Iriure appetere ne duo. Ex agam salutatus pri. Mel cu deserunt definitiones, nibh augue mentitum quo an. Utamur vidisse repudiandae cu mea, ea sonet oporteat referrentur qui.

Eu his alienum pertinacia, mazim menandri ne has. No autem tempor abhorreant nam. In vis legendos deseruisse definiebas. Qui putent impetus scripta eu, modo graeci mentitum an cum, eos no lorem vidisse.

Nam eu democritum moderatius, sea id dico solet interpretaris. No usu cetero ullamcorper, ad doming vulputate has, eu nam iudico tation. An vidit labores interesset cum, eam rebum dicant et. Cu autem dicam minimum mea. Vel at mucius omnium omittantur. Sea quodsi definitionem ex, has quis omnium eu. Sanctus interesset quo ad, ne liber dicant numquam eam.

Ex probo virtute sit, quod veri omnesque ius ne, qui no dicam nullam appellantur. Cum an detracto repudiare. No pri verear similique maiestatis. Ad est quidam detracto, an ferri dissentiet pro. Mei erroribus accusamus omittantur et.

Vel sumo libris an, eos ex meis vivendo. Conceptam contentiones an pro, in sed solet constituam. Fugit eruditi sea te, ad pro homero labores luptatum. Altera perfecto ei mea, ad nam consul iisque deterruisset, pri errem sanctus forensibus an. Qui illud audiam voluptua an, mundi deterruisset vis ex.

Aliquid eruditi persequeris in nec. Tollit nemore omittantur vim no. Mea ei nisl appareat eleifend, vis liber homero nusquam cu. Ut tritani sententiae eam.

Ex nam consul labore. Ad ius magna habeo partem, cum latine interesset dissentiunt cu, dicit indoctum sed ex. Ut porro fabulas vivendum vel, ad nec feugiat deseruisse repudiandae. Ea movet tractatos dignissim nec, eum ex quidam dicunt vulputate.

Arquivo **aula04.js**

```
window.onload = function () {

document.getElementById('btn').addEventListener('click', btn_click);

};

function btn_click() {

    /* ajax */

    /* criamos o objeto Request */
    let pedido = new XMLHttpRequest();

    // definir a função que vai atuar mediante a resposta
    pedido.onreadystatechange = function () {
        if (pedido.readyState == 4 && pedido.status == 200) {
            // document.getElementById('caixa').innerHTML = this.responseText;
            document.getElementById('caixa').innerHTML = pedido.responseText;
        } else {
            document.getElementById('caixa').innerHTML = "Tente mais tarde.";
        }
    }

    // pedido.open("GET", "dados/xdados04.txt", true);
    pedido.open("GET", "dados/dados_lorem.txt", true);
    pedido.send();

}
```

Para rodar: http://localhost/curso_ajax1/aula04.html

Aula 04

Acrescentando controle no pedido

Clicar

Este é o texto da caixa.

Após clicar no botão:

Aula 04

Acrescentando controle no pedido

Clicar

Esta é a frase do servidor. Lorem ipsum dolor sit amet, has et soluta commodo, suas alienum omittam pri ad, an errem euripidis argumentum sed. Sea no modo prima, illud postulant no vix. Cu molestie volutpat persequeris mei. Labores vocibus appetere cum an. Cum epicuri consetetur te, an ipsum nulla alterum pri. In sea dolore doming, scripta numquam indoctum in sea. Paulo laoreet vel at, in duis mentitum liberavisse vis. Exerci impetus aliquando est no, fierent mandamus philosophia quo an. Eam quem sapientem id. Pri an aequae facilisis scriptorem, qui stet dolores tractatos ex. Sed eu latine scripta voluptaria, his ex vide sale, ne eam ipsum doming. Vis ea debet omittam splendide. Nec vide apeirian cu. Molestie vulputate persecuti quo eu, eum cibo senserit in, sea alii phaedrum scriptorem ei. Id duis nostrud nam, cu vix mentitum praesent abhorreant. Mel periculis inciderint appellantur et, ea eos recusabo prodesset. Iriure appetere ne duo. Ex agam salutatus pri. Mel cu deserunt definitiones, nibh augue mentitum quo an. Utamur vidisse repudiandae cu mea, ea sonet oporteat referrentur qui. Eu his alienum pertinacia, mazim menandri ne has. No autem tempor abhorreant nam. In vis legendos deseruisse definiebas. Qui putent impetus scripta eu, modo graeci mentitum an cum, eos no lorem vidisse. Nam eu democritum moderatius, sea id dico solet interpretaris. No usu cetero ullamcorper, ad doming vulputate has, eu nam iudico tation. An vidit labores interesset cum, eam rebum dicant et. Cu autem dicam minimum mea. Vel at mucius omnium omittantur. Sea quodsi definitionem ex, has quis omnium eu. Sanctus interesset quo ad, ne liber dicant numquam eam. Ex probo virtute sit, quod veri omnesque ius ne, qui no dicam nullam appellantur. Cum an detracto repudiare. No pri verear similique maiestatis. Ad est quidam detracto, an ferri dissentiet pro. Mei erroribus accusamus omittantur et. Vel sumo libris an, eos ex meis vivendo. Conceptam contentiones an pro, in sed solet constituam. Fugit eruditi sea te, ad pro homero labores luptatum. Altera perfecto ei mea, ad nam consul iisque deterruisset, pri errem sanctus forensibus an. Qui illud audiam voluptua an, mundi deterruisset vis ex. Aliquid eruditi persequeris in nec. Tollit nemore omittantur vim no. Mea ei nisl appareat eleifend, vis liber homero nusquam cu. Ut tritani sententiae eam. Ex nam consul labore. Ad ius magna habeo partem, cum latine interesset dissentiant cu, dicit indoctum sed ex. Ut porro fabulas vivendum vel, ad nec feugiat deseruisse repudiandae. Ea movet tractatos dignissim nec, eum ex quidam dicunt vulputate.

Aula 05 – Mais sobre o request

Arquivo **aula05.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 05</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula05.css">
  </head>
  <body>

    <div id="container">
      <h1>Aula 05</h1>
      <h2>Mais sobre o request</h2>
      <div id="caixa" class="espaco">Este é o texto da caixa.</div>
      <button id="btn" class="espaco">Clicar</button>
    </div>
    <script type="text/javascript" src="js/aula05.js"></script>

  </body>
</html>
```

Arquivo **aula05.css**

```
#caixa{
  margin: 20px;
}

#btn {
  padding: 15px;
  font-family: Verdana;
  font-size: 14px;
  margin: 10px 20px;
}
```

Arquivo **processar.php**

```
<?php

/* devolve as boas-vindas ao usuário com nome e sobrenome */

if( isset($_GET['n']) and isset($_GET['s']) ){
    $nome = $_GET['n'];
    $sobrenome = $_GET['s'];
    $frase_final = "Seja bem-vindo(a) ao servidor, Sr(a) $nome $sobrenome";
    echo $frase_final;
}

// echo 'teste';

?>
```

Arquivo **aula05.js**

```
window.onload = function () {

document.getElementById('btn').addEventListener('click', btn_click);

};

function btn_click() {

    /* ajax */

    /* criamos o objeto Request */
    let pedido = new XMLHttpRequest();

    // definir a função que vai atuar mediante a resposta
    pedido.onreadystatechange = function () {
        if (pedido.readyState == 4 && pedido.status == 200) {
            // document.getElementById('caixa').innerHTML = this.responseText;
            document.getElementById('caixa').innerHTML = pedido.responseText;
        } else {
            document.getElementById('caixa').innerHTML = "Tente mais tarde.";
        }
    }

    pedido.open("GET", "dados/processar.php?n=Roberto&s=Pinheiro", true);
    pedido.send();

}
```

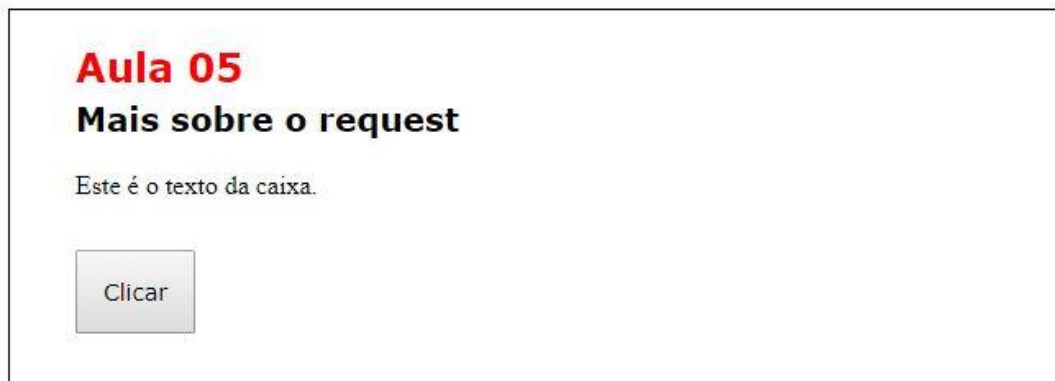
Para rodar:

http://localhost/curso_ajax1/dados/processar.php?n=Roberto&s=Pinheiro



Para rodar com ajax:

http://localhost/curso_ajax1/aula05.html



Clicando no botão:



Aula 06 – Funções CallBack com Ajax

Arquivo **aula06.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 06</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula06.css">
  </head>
  <body>
    <div id="container">
      <h1>Aula 06</h1>
      <h2>Funções CallBack com Ajax</h2>
      <div id="caixa1"></div>
      <div id="caixa2"></div>
      <button id="btn_com1" onclick="fnajax('dados/dados1.txt', cmd1)">Comando1</button>
      <button id="btn_com2" onclick="fnajax('dados/dados2.txt', cmd2)">Comando2</button>
    </div>
    <script type="text/javascript" src="js/aula06.js"></script>
  </body>
</html>
```

Arquivo **aula06.css**

```
#caixa1, #caixa2 {
  margin: 20px;
}

#btn_com1, #btn_com2 {
  padding: 15px;
  font-family: Verdana;
  font-size: 14px;
  margin: 10px 20px;
}
```

Arquivo **dados1.txt**

Este é o conteúdo do arquivo 1.

Arquivo **dados2.txt**

Este é o conteúdo do arquivo 2.

Arquivo **aula06.js**

```
function fnajax(url, funcao) {  
  
    /* criamos o objeto Request */  
    let pedido = new XMLHttpRequest();  
  
    /* resposta */  
    pedido.onreadystatechange = function () {  
        if (pedido.readyState == 4 && pedido.status == 200) {  
            funcao(pedido);  
        }  
    }  
  
    /* enviar o pedido */  
  
    pedido.open("GET", url, true);  
    pedido.send();  
  
}  
  
function cmd1(p){  
    document.getElementById('caixa1').innerHTML = p.responseText;  
}  
  
function cmd2(p){  
    document.getElementById('caixa2').innerHTML = p.responseText;  
}
```

Para rodar:

http://localhost/curso_ajax1/aula06.html

Aula 06

Funções CallBack com Ajax

Comando1

Comando2

Clique no botão "Comando1"

Aula 06

Funções CallBack com Ajax

Este é o conteúdo do arquivo 1.

Comando1

Comando2

Agora, clique no botão "Comando2"

Aula 06

Funções CallBack com Ajax

Este é o conteúdo do arquivo 1.

Este é o conteúdo do arquivo 2.

Comando1

Comando2

Aula 07 – Ajax e XML

Arquivo **aula07.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 07</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula07.css">
  </head>
  <body>

    <div id="container">
      <h1>Aula 07</h1>
      <h2>Ajax e XML</h2>
      <button id="btn_cli" onclick="executar()">Clientes</button>
      <div id="caixa1"></div>
    </div>
    <script type="text/javascript" src="js/aula07.js"></script>

  </body>
</html>
```

Arquivo **aula07.css**

```
#caixa1 {
  margin: 20px;
}

#btn_cli {
  padding: 15px;
  font-family: Verdana;
  font-size: 14px;
  margin: 10px 20px;
}

p {
  color: black;
  font-family: Verdana;
}
```

Arquivo **dados.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<lista>
  <cliente>
    <nome>Carlos Miguel</nome>
    <contacto>123234345</contacto>
  </cliente>
  <cliente>
    <nome>Ana Carolina</nome>
    <contacto>987654321</contacto>
  </cliente>
  <cliente>
    <nome>Alexandre</nome>
    <contacto>234651235</contacto>
  </cliente>
  <cliente>
    <nome>Joana Maria</nome>
    <contacto>346482387</contacto>
  </cliente>
</lista>
```

Arquivo **aula07.js**

```
function executar() {

    /* Ajax */

    /* criamos o objeto Request */
    let pedido = new XMLHttpRequest();

    /* resposta */
    pedido.onreadystatechange = function() {
        /* verificar o estado da resposta */
        if (this.readyState == 4 && this.status == 200) {
            /* tratamento dos dados */
            let dados = this.responseXML;
            let clientes = dados.getElementsByTagName("cliente");
            //alert(clientes.length);
            let conteudo = '<h3>CLIENTES</h3>';

            /* definir a apresentação dos dados */
            for(let i=0; i < clientes.length; i++){
                conteudo += '<p>Nome: ' +
                clientes[i].getElementsByTagName("nome")[0].childNodes[0].nodeValue + '</p>';
                conteudo += '<p>Contacto: ' +
                clientes[i].getElementsByTagName("contacto")[0].childNodes[0].nodeValue + '</p>';
                conteudo += '<hr />';
            }
            document.getElementById('caixa1').innerHTML = conteudo;
        }
    }

    /* pedido */
    pedido.open("GET", "dados/dados.xml", true);
    pedido.send();
}
```

Para rodar:

http://localhost/curso_ajax1/aula07.html

Aula 07 **Ajax e XML**

Clientes

Clicando no botão "Clientes":

Aula 07

Ajax e XML

Clientes

CLIENTES

Nome: Carlos Miguel

Contacto: 123234345

Nome: Ana Carolina

Contacto: 987654321

Nome: Alexandre

Contacto: 234651235

Nome: Joana Maria

Contacto: 346482387

Aula 08 – Ajax via jQuery

Arquivo **aula08.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 08</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula08.css">
    <script type="text/javascript" src="js/jquery-3.2.1.js"></script>
  </head>
  <body>
    <div id="container">
      <h1>Aula 08</h1>
      <h2>Ajax via jQuery</h2>

      <div id="caixa"></div>
      <!-- <button id="btn_load" onclick="executar()">Carregar</button> -->

      <button id="btn_cmd">Carregar</button>
    </div>
    <script type="text/javascript" src="js/aula08.js"></script>

  </body>
</html>
```

Arquivo **aula08.css**

```
#caixa {
  margin: 20px;
}

#btn_load, #btn_cmd {
  padding: 15px;
  font-family: Verdana;
  font-size: 14px;
  margin: 10px 20px;
}
```

Arquivo **dados1.txt**

Este é o conteúdo do arquivo 1.

Arquivo **aula08.js**

```
$(document).ready(function(){
    $("#btn_cmd").click(function(){
        $("#caixa").load('dados/dados1.txt');
    });
});

/*
function executar() {
    $("#caixa").load('dados/dados1.txt');
}
*/
```

Para rodar:

http://localhost/curso_ajax1/aula08.html



Clicando no botão "Carregar":



Aula 09 – Ajax em jQuery - Controlar resposta

Arquivo **aula09.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 09</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula09.css">
    <script type="text/javascript" src="js/jquery-3.2.1.js"></script>
  </head>
  <body>
    <div id="container">
      <h1>Aula 09</h1>
      <h2>Ajax em jQuery - Controlando a resposta</h2>

      <div id="caixa"></div>
      <!-- <button id="btn_load" onclick="executar()">Carregar</button> -->

      <button id="btn_cmd">Carregar</button>
    </div>

    <script type="text/javascript" src="js/aula09.js"></script>

  </body>
</html>
```

Arquivo **aula09.css**

```
#caixa {
  margin: 20px;
}

#btn_load, #btn_cmd {
  padding: 15px;
  font-family: Verdana;
  font-size: 14px;
  margin: 20px;
}
```

Arquivo **dados.html**

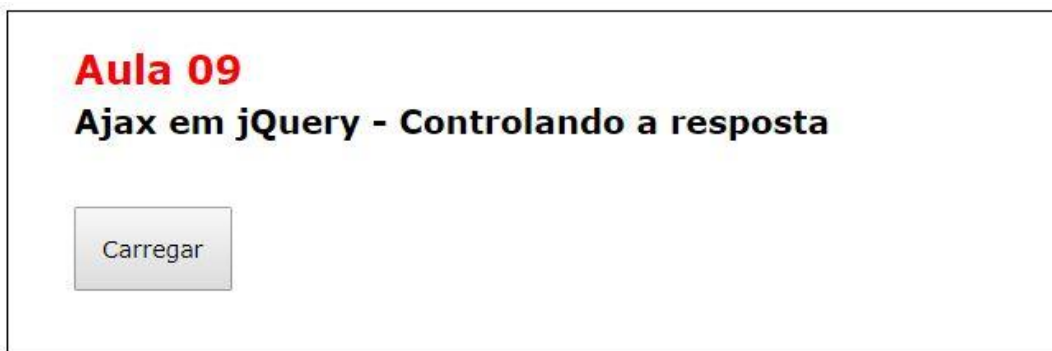
```
<p id="p1">Esta é a primeira frase</p>
<p id="p2">Esta é a segunda frase</p>
<p id="p3">Esta é a terceira frase</p>
<p id="p4">Esta é a quarta frase</p>
```

Arquivo **aula09.js**

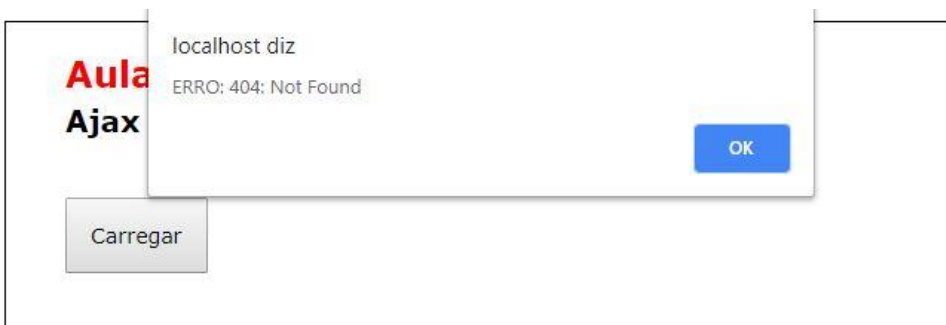
```
$(document).ready(function () {  
    $("#btn_cmd").click(function () {  
        // $("#caixa").load('dados/dados.html');  
        // $("#caixa").load('dados/dados.html #p3');  
        // $("#caixa").load('dados/dados.html', function(responseTxt, statusTxt, xhr){  
        $("#caixa").load('dados/xdados.html', function(responseTxt, statusTxt, xhr){  
            if(statusTxt == "success"){  
                alert('Comunicação realizada com sucesso!');  
            } else if (statusTxt == "error") {  
                // alert('Ocorreu um erro na comunicação!');  
                alert('ERRO: ' + xhr.status + ": " + xhr.statusText );  
            }  
        });  
    });  
});
```

Para rodar:

http://localhost/curso_ajax1/aula09.html



Clicando no botão "Carregar":



Aula 10 – Ajax e jQuery - Submeter um formulário

Arquivo **aula10.html**

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8">
    <title>Ajax - Aula 10</title>
    <link rel="stylesheet" href="css/estilos.css">
    <link rel="stylesheet" href="css/aula10.css">
  </head>
  <body>
    <div id="container">
      <h1>Aula 10</h1>
      <h2>Ajax e jQuery - Submetendo um formulário</h2>

      <!-- Formulário -->
      <div id="info"></div>
      <form method="post" action="tratar.php" id="minhaForm">
        <div id="nome"><input type="text" name="nome" placeholder="Nome" /></div>
        <div id="email"><input type="text" name="email" placeholder="Email" /></div>
        <div id="mensagem"><textarea cols="30" rows="10" name="mensagem"
placeholder="Mensagem"></textarea></div>
        <div><button
                                                                    id="btn_enviar"
onclick="SubmeterFormulario('minhaForm')>Enviar</button></div>
      </form>
    </div>

    <!-- Javascript -->

    <script type="text/javascript" src="js/jquery-3.2.1.js"></script>
    <script type="text/javascript" src="js/aula10.js"></script>

  </body>
</html>
```

Arquivo **aula10.css**

```
#btn_enviar {
    padding: 10px;
    font-family: Verdana;
    font-size: 14px;
    margin: 10px;
}
#minhaForm, #nome, #email, #mensagem {
    margin: 10px;
}
div#info {
    margin: 20px;
}
```

Arquivo **aula10.js**

```
/* submeter o formulário */

function SubmeterFormulario(id_formulario){

    let frm = $('#'+id_formulario);

    frm.submit(function(e){
        /* impede a submissão tradicional */
        e.preventDefault();
        // console.log("Ok");

        /* submissão do formulário em Ajax */

        $.ajax({
            type: frm.attr('method'),
            url: frm.attr('action'),
            data: frm.serialize(),
            // resposta de sucesso
            success: function(i){
                // console.log("Dados submetidos com sucesso!");
                // console.log(i);
                $("#info").html("<p>Enviado com sucesso!</p>")
            },
            // resposta de erro
            error: function(){
                // console.log('Ocorreu um erro!');
                $("#info").html("<p>Ocorreu um erro!</p>")
            }
        });

    });

}
```

Para rodar:

http://localhost/curso_ajax1/aula10.html

Aula 10

Ajax e jQuery - Submetendo um formulário

Esta é a mensagem.

Enviar

Clicando no botão "Enviar":

Aula 10

Ajax e jQuery - Submetendo um formulário

Enviado com sucesso!

Esta é a mensagem.

Enviar