

```
//AJAX:

//Devemos lembrar sempre que a grande vantagem de uma requisição AJAX é fazer requisição URL e carregá-la na página sem precisar de
fazer um refresh.

//Podemos ver esse comportamento facilmente em páginas onde colocamos o CEP e o formulário auto-completa para nós, isso é possível por
através de uma requisição AJAX.

//Seria muito simples implementar uma requisição AJAX para CEP usando JQuery, mas teríamos que importar toda uma biblioteca somente
para isso. Por isso é importante que aprendamos a fazer requisições AJAX independentes de uma biblioteca.

//Vejam os abaixo como fazer essa requisição usando XHR (XMLHttpRequest):

//No nosso formulário estamos usando um botão que não é para submissão, ele tem como objetivo pesquisar o endereço de um CEP, como os
button tem por padrão o evento de submissão recarrega a página atual, devemos evitar que o nosso botão execute esse evento, para isso
criamos a função "eviteSubmit"...
function evitaSubmit(evite){evite.preventDefault()}

//Abaixo temos um comando para botão que retira o evento "submit" usando a função criada anteriormente e adiciona o evento
"click"...
document.getElementById("o-button--gary").addEventListener('click', evitaSubmit, false)

//Agora vamos a parte que interessa a criação de 2 funções, uma para pesquisar o CEP e outra para preencher os campos do nosso
formulário automaticamente.

//1ª Precisamos criar uma função que vai encapsular todo o processo de buscar um CEP, para isso vamos criar a função "buscaCEP":

function buscaCEP(){
    //O primeiro passo dessa função deve ser pegar a referência do campo CEP para obtermos seu valor, podemos fazer isso por através
da variável inputCep logo abaixo:
    let inputCep = document.querySelector('input[name="cep"]');

    //Podemos criar outra variável responsável por pegar somente o valor de CEP e limpá-lo para que possamos usar na URL
    let cep = inputCep.value.replace('-', ''); //O replace serve para omitir o "traço" caso o cliente digite ele, pois
                                                //só podemos usar números na url do viacep...
```

```

    //Próximo passo é construir a url que vá receber o nosso cep retornar para nós um JSON com todos os detalhes relevantes daquele CEP...
    let url = 'http://viacep.com.br/ws/' + cep + '/json';

    //Mas como o javascript vai fazer esse requisição de CEP?
    //Para isso o Javascript possui uma classe nativa que realiza requisições AJAX conhecida como XMLHttpRequest. Apesar do nome, ele pode ser usada para recuperar qualquer tipo de dado e fornece suporte aos protocolos HTTP, FILE e FTP...
    let xhr = new XMLHttpRequest() //o nome xhr da variável é uma referência a XMLHttpRequest...

    //Com o objeto XHR criado, vamos utilizar o método open para começarmos a criar a nossa requisição:
    xhr.open('GET', url, true) /*O método open recebe três parâmetros, sendo eles:
        1º - O verbo HTTP a ser utilizado para realizar a requisição, o mesmo deve seguir o padrão REST, o verbo que usaremos no caso é "GET";
        2º - A URL que pretendemos obter os dados.
        3º - Um argumento booleano informando se a requisição deve ser assíncrona ou síncrona, "true" para assíncrona e "false" para síncrona.*/

    //Com isso já criamos uma requisição GET para a nossa URL, mas além disso é importante usarmos o método "onreadystatechange" para disparar sempre que nossa requisição sofre alguma alteração durante seu processamento:
    xhr.onreadystatechange = function() { //Repare que dentro da chamada há uma callback que vai ser disparada a cada
        if (xhr.readyState == 4){ //nova alteração. E ela executa 2 condicionais onde se "readyState" tiver valor 4,
            if (xhr.status == 200){ //significa que a requisição foi finalizada, entrando na próxima condicional "if",
                preencheCampos(JSON.parse(xhr.responseText)) //que usa a chave "status" para nos informar se a
            } //requisição teve sucesso ou não, se o valor for igual a "200"
        } //significa que foi bem sucedida. Então ela vai chamar a função
    } // "preencheCampos" que irá receber o arquivo JSON com os detalhes
    //do CEP, e irá convertê-los para um objeto usando o JSON.parse...

    xhr.send()
}

//Aqui temos a função preencheCampos que recebe o objeto que foi convertido do JSON advindo da URL e atribuí cada valor chave ao respectivo campo no formulário, fazendo o auto-preenchimento...
function preencheCampos(json) {
    document.querySelector('input[name=endereco]').value = json.logradouro
    document.querySelector('input[name=bairro]').value = json.bairro
    document.querySelector('input[name=complemento]').value = json.complemento
}

```

```
}
    document.querySelector('input[name=cidade]').value = json.localidade
    document.querySelector('input[name=estado]').value = json.uf
}
```

Arquivo HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Ajax com JS Puro</title>
  <style>
    button {
      height: 20px;
      background-color: green;
      color: white;
    }
  </style>
</head>
<body>
  <form class="o-form">
    <div class="o-navbar--gary">Buscando CEP com requisição AJAX</div>
    <div class="o-form__body o-form__body--padding">
      <div class="l-row">
        <div class="u-grid--2">
          <label for="" class="o-form__text">CEP</label>
          <div class="o-form__group">
            <input class="o-form__data" name="cep" type="text">
            <span class="o-form__groupElement">
              <button id="o-button--gary" onclick="buscaCEP()"> <!--Aqui temos o evento chamando a -->
                Pesquisar <!--função "buscaCEP()..."-->
              </button>
            </span>
          </div>
        </div>
      </div>
    </div>
  </form>
</body>
</html>
```

```
<div class="u-grid--8">
  <label for="" class="o-form__text">Endereço</label>
  <input class="o-form__data" name="endereco" type="text">
</div>
<div class="u-grid--2">
  <label for="" class="o-form__text">Numero</label>
  <input class="o-form__data" name="numero" type="text">
</div>
</div>
<div class="l-row">
  <div class="u-grid--8">
    <label for="" class="o-form__text">Bairro</label>
    <input class="o-form__data" name="bairro" type="text">
  </div>
  <div class="u-grid--4">
    <label for="" class="o-form__text">Complemento</label>
    <input class="o-form__data" name="complemento" type="text">
  </div>
</div>
<div class="l-row">
  <div class="u-grid--10">
    <label for="" class="o-form__text">Cidade</label>
    <input class="o-form__data" name="cidade" type="text">
  </div>
  <div class="u-grid--2">
    <label for="" class="o-form__text">Estado</label>
    <input class="o-form__data" name="estado" type="text">
  </div>
</div>
</div>
</form>
<script type="text/javascript" src="197-AJAX___Fazendo_requisicoes_HTML_sem_recarregar_a_pagina.js"></script>
</body>
</html>
```

RESULTADO NO NAVEGADOR:

1ª Pesquisa:

Buscando CEP com requisição AJAX

CEP

Endereço

Numero

Bairro

Complemento

Cidade

Estado

2ª Pesquisa:

Buscando CEP com requisição AJAX

CEP

Endereço

Numero

Bairro

Complemento

Cidade

Estado